NI Multisim™

User Manual

January 2009
374483D-01

NATIONAL INSTRUMENTS™
ELECTRONICS WORKBENCH GROUP

Worldwide Technical Support and Product Information

ni.com

National Instruments Corporate Headquarters

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 683 0100

Worldwide Offices

Australia 1800 300 800, Austria 43 662 457990-0, Belgium 32 (0) 2 757 0020, Brazil 55 11 3262 3599, Canada 800 433 3488, China 86 21 5050 9800, Czech Republic 420 224 235 774, Denmark 45 45 76 26 00, Finland 358 (0) 9 725 72511, France 01 57 66 24 24, Germany 49 89 7413130, India 91 80 41190000, Israel 972 3 6393737, Italy 39 02 41309277, Japan 0120-527196, Korea 82 02 3451 3400, Lebanon 961 (0) 1 33 28 28, Malaysia 1800 887710, Mexico 01 800 010 0793, Netherlands 31 (0) 348 433 466, New Zealand 0800 553 322, Norway 47 (0) 66 90 76 60, Poland 48 22 328 90 10, Portugal 351 210 311 210, Russia 7 495 783 6851, Singapore 1800 226 5886, Slovenia 386 3 425 42 00, South Africa 27 0 11 805 8197, Spain 34 91 640 0085, Sweden 46 (0) 8 587 895 00, Switzerland 41 56 2005151, Taiwan 886 02 2377 2222, Thailand 662 278 6777, Turkey 90 212 279 3031, United Kingdom 44 (0) 1635 523545

For further support information, refer to the *Technical Support and Professional Services* appendix. To comment on National Instruments documentation, refer to the National Instruments Web site at ni.com/info and enter the info code feedback.

© 2006–2009 National Instruments Corporation. All rights reserved.

Important Information

Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this document is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

National Instruments respects the intellectual property of others, and we ask our users to do the same. NI software is protected by copyright and other intellectual property laws. Where NI software may be used to reproduce software or other materials belonging to others, you may use NI software only to reproduce materials that you may reproduce in accordance with the terms of any applicable license or other legal restriction.

BSIM3 and BSIM4 are developed by the Device Research Group of the Department of Electrical Engineering and Computer Science, University of California, Berkeley and copyrighted by the University of California.

The ASM51 cross assembler bundled with Multisim MCU is a copyrighted product of MetaLink Corp. (`www.metaice.com`).

MPASM™ macro assembler and related documentation and literature is reproduced and distributed by Electronics Workbench under license from Microchip Technology Inc. All rights reserved by Microchip Technology Inc. MICROCHIP SOFTWARE OR FIRMWARE AND LITERATURE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL MICROCHIP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY ARISING OUT OF OR IN CONNECTION WITH THE SOFTWARE OR FIRMWARE OR THE USE OF OTHER DEALINGS IN THE SOFTWARE OR FIRMWARE.

Anti-Grain Geometry - Version 2.4

Copyright (C) 2002-2004 Maxim Shemanarev (McSeem)

Permission to copy, use, modify, sell and distribute this software is granted provided this copyright notice appears in all copies. This software is provided "as is" without express or implied warranty, and with no claim as to its suitability for any purpose.

Anti-Grain Geometry - Version 2.4

Copyright (C) 2002-2005 Maxim Shemanarev (McSeem)

1. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

2. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

3. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Trademarks

National Instruments, NI, ni.com, and LabVIEW are trademarks of National Instruments Corporation. Refer to the *Terms of Use* section on `ni.com/legal` for more information about National Instruments trademarks.

Electronics Workbench, Multisim and Ultiboard are trademarks of National Instruments.

Portions of this product obtained under license from Bartels Systems GmbH.

Other product and company names mentioned herein are trademarks or trade names of their respective companies.

Members of the National Instruments Alliance Partner Program are business entities independent from National Instruments and have no agency, partnership, or joint-venture relationship with National Instruments.

Patents

For patents covering National Instruments products/technology, refer to the appropriate location: **Help»Patents** in your software, the `patents.txt` file on your media, or the *National Instruments Patent Notice* at `ni.com/patents`.

Some portions of this product are protected under United States Patent No. 6,560,572.

WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

(1) NATIONAL INSTRUMENTS PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED “SYSTEM FAILURES”). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM NATIONAL INSTRUMENTS' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE NATIONAL INSTRUMENTS PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY NATIONAL INSTRUMENTS, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF NATIONAL INSTRUMENTS PRODUCTS WHENEVER NATIONAL INSTRUMENTS PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

Conventions

The following conventions are used in this manual:

»	The » symbol leads you through nested menu items and dialog box options to a final action. The sequence **File»Page Setup»Options** directs you to pull down the **File** menu, select the **Page Setup** item, and select **Options** from the last dialog box.
	This icon denotes a tip, which alerts you to advisory information.
	This icon denotes a note, which alerts you to important information.
	This icon denotes a caution, which advises you of precautions to take to avoid injury, data loss, or a system crash.
bold	Bold text denotes items that you must select or click in the software, such as menu items and dialog box options. Bold text also denotes parameter names.
italic	Italic text denotes variables, emphasis, a cross-reference, or an introduction to a key concept. Italic text also denotes text that is a placeholder for a word or value that you must supply.
`monospace`	Text in this font denotes text or characters that you should enter from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames, and extensions.
`monospace bold`	Bold text in this font denotes the messages and responses that the computer automatically prints to the screen. This font also emphasizes lines of code that are different from the other examples.
`monospace italic`	Italic text in this font denotes text that is a placeholder for a word or value that you must supply.

Contents

Chapter 1
User Interface

Introduction to the Multisim Interface 1-1
Toolbars 1-1
- Standard Toolbar 1-2
- Main Toolbar 1-3
- Simulation Toolbar 1-4
- View Toolbar 1-5
- Components Toolbar 1-6
- Virtual Toolbar 1-7
- Graphic Annotation Toolbar 1-7
- Instruments Toolbar 1-8
- Locking Toolbars 1-11
- Toolbar Behavior When Multisim Size is Reduced 1-12

Pop-up Menus 1-12
- Pop-up From Circuit Window, with no Component Selected 1-12
- Pop-up From a Selected Component or Instrument 1-15
- Pop-up From a Selected Wire 1-17
- Pop-up From a Selected Text Block or Graphic 1-17
- Pop-up From a Title Block 1-18
- Pop-up from a Comment or Measurement Probe 1-19

Setting Schematic Capture Preferences 1-19
- Using the Preferences Dialog Box 1-20
  - Preferences - Paths Tab 1-20
  - Preferences - Save Tab 1-21
  - Preferences - Parts Tab 1-22
  - Preferences - General Tab 1-23
  - Preferences - Simulation Tab 1-23
- Using the Sheet Properties Dialog Box 1-24
  - Sheet Properties - Circuit Tab 1-25
  - Sheet Properties - Workspace Tab 1-26
  - Sheet Properties - Wiring Tab 1-27
  - Sheet Properties - Font Tab 1-27
  - Sheet Properties - PCB Tab 1-28
  - Sheet Properties - Visibility Tab 1-29

Design Toolbox 1-29
- Visibility Tab 1-29
- Hierarchy Tab 1-30

Customizing the Interface 1-32
Commands tab 1-33
Toolbars tab 1-33
Keyboard tab 1-34
Menu tab 1-34
Options tab 1-35
Customization Pop-up Menus 1-35

Chapter 2
Schematic Capture - Basics

Introduction to Schematic Capture 2-1
Working with Multiple Circuit Windows 2-1
Selecting Components from the Database 2-2
Placing Components 2-2
Using the Place Component Browser 2-3
Placing Resistors, Inductors or Capacitors 2-5
Multi-section Components 2-8
Rotating/flipping a part during placement 2-10
Other Buttons 2-11
Placing Virtual Components 2-11
Virtual component toolbars 2-13
Using the In Use List 2-14
Drop-Placing Two-Pinned Components 2-15
Selecting Placed Components 2-15
Moving a Placed Component 2-16
Copying a Placed Component 2-18
Replacing a Placed Component 2-18
Controlling Component's Color 2-19
Wiring Components 2-19
Wiring Components Automatically 2-20
Autowire of Touching Pins 2-22
Wiring Components Manually 2-22
Combining Automatic and Manual Wiring 2-23
Marking Pins for No Connection 2-23
Placing Wires Directly Onto Workspace 2-24
Setting Wiring Preferences 2-24
Modifying the Wire Path 2-25
Controlling Wire Color 2-26
Moving a Wire 2-26
Virtual Wiring 2-27
Manually Adding a Junction 2-27
Rotating/Flipping Placed Components 2-28
Finding Components in Your Circuit 2-29

Labeling 2-31
Modifying Component Labels and Attributes 2-31
Modifying Net Names 2-32
Adding a Title Block 2-33
Entering the Title Block Contents 2-33
Adding Miscellaneous Text 2-35
Adding a Comment 2-36
Graphic Annotation 2-38
Capturing Screen Area 2-41
Circuit Description Box 2-42
Formatting the Circuit Description Box 2-43
Formatting Circuit Description Box Text 2-44
Paragraph Dialog Box 2-44
Tabs Dialog Box 2-45
Date and Time Dialog Box 2-45
Properties Dialog Box 2-46
Insert Object Dialog Box 2-46
Scrolling with Events During Simulation 2-47
Scrolling Text During Simulation 2-47
Playing a Video Clip 2-49
Description Label Dialog Box 2-50
Edit Labels Dialog Box 2-51
Other Actions 2-52
Description Edit Bar 2-52
Linking a Form to a Circuit 2-54
Creating Forms 2-54
Linking to Questions 2-56
Setting Form Submission Options 2-56
Completing Forms 2-57
Printing the Circuit 2-58

Chapter 3
Schematic Capture - Advanced Functions

Placed Component Properties 3-1
Displaying Identifying Information about a Placed Component 3-1
Viewing a Placed Component's Value/Model 3-2
Real Components 3-2
Resistors, Inductors and Capacitors 3-3
Edit Model Dialog Box 3-4
Edit Footprint Dialog Box 3-5
Virtual Components 3-5
Controlling How a Placed Component is Used in Analyses 3-6
Editing a Placed Component's User Fields 3-7

Assigning Faults to Components 3-8
Setting a Placed Component's Faults 3-8
Using the Auto Fault Option 3-9
Spreadsheet View 3-10
Spreadsheet View Results Tab 3-10
Spreadsheet View Nets Tab 3-10
Spreadsheet View Components Tab 3-12
Spreadsheet View PCB Layers Tab 3-14
Spreadsheet View Simulation Tab 3-14
Spreadsheet View Buttons 3-15
Title Block Editor 3-16
Enter Text Dialog Box 3-19
Placing Fields 3-19
Field Codes 3-21
Title Block Editor Spreadsheet View 3-22
Title Block Editor Menus 3-23
File Menu 3-23
Edit Menu 3-23
View Menu 3-24
Fields Menu 3-25
Graphics Menu 3-27
Tools Menu 3-28
Help Menu 3-28
Pop-up Menus 3-28
Title Block Editor Toolbars 3-28
Standard Toolbar - Title Block Editor 3-29
Zoom Toolbar - Title Block Editor 3-30
Draw Tools Toolbar - Title Block Editor 3-30
Drawing Toolbar - Title Block Editor 3-32
Electrical Rules Checking 3-34
ERC Options Tab 3-37
Clearing ERC Markers 3-38
ERC Rules Tab 3-39
Component's Pins Tab 3-41

Chapter 4
Working with Larger Designs

Flat Multi-Sheet Design 4-1
Delete Multi-Page Dialog Box 4-2
Hierarchical Design 4-2
Nested Circuits 4-3
Component Numbering in Nested Circuits 4-4
Net Numbering in Nested Circuits 4-5

Global Nets 4-5
Adding a Hierarchical Block 4-7
Placing a HB from an Existing File 4-8
Replacing Components with an HB 4-9
Adding a Subcircuit 4-10
Replacing Components with an SC 4-10
Viewing a Parent Sheet 4-11
Renaming Component Instances 4-11
Reference Designator Prefix Setup Dialog 4-13
Buses 4-15
Placing a Bus 4-18
Placing a Bus across Multi-Pages 4-18
Connecting Buses to HB/SCs 4-19
Bus Properties 4-21
Adding Buslines to a Bus 4-21
Deleting Buslines from a Bus 4-21
Renaming Buslines in a Bus 4-21
Merging Buses 4-22
Wiring to a Bus 4-23
Bus Vector Connect 4-26
Variants 4-37
Setting Up Variants 4-38
Placing Parts in Variants 4-41
Assigning Variant Status to Components 4-42
Assigning Variant Status to Nested Circuits 4-47
Setting the Active Variant for Simulation 4-47
Project Management and Version Control 4-50
Setting up Projects 4-50
Working with Projects 4-51
Working with Files Contained in Projects 4-52
Version Control 4-53

Chapter 5
Components

Structure of the Component Database 5-1
Database Levels 5-2
Classification of Components in the Database 5-2
Locating Components in the Database 5-3
Browsing for Components 5-3
Searching for Components 5-3
Types of Information Stored for Components 5-4
Pre-Defined Fields 5-5
User Fields 5-5

Managing the Database 5-6
Filtering Displayed Components 5-6
Deleting Components 5-7
Copying Components 5-7
Exporting Components 5-8
Saving Placed Components 5-8
Moving Components Between Databases 5-9
Resetting Master dB User Fields 5-9
Managing Families 5-9
Modifying User Field Titles and Content 5-10
Modifying User Field Content for RLC Components 5-11
Displaying Database Information 5-12
Editing Components 5-12
Converting Databases 5-12
Updating Components from Databases 5-14
Merging Databases 5-16

Chapter 6
Component Editing

Introduction to Component Editing 6-1
Creating Components with the Component Wizard 6-3
Creating an Analog Component 6-3
Creating a Digital Component 6-7
Creating a VHDL Component 6-11
Using a Symbol File Created in the Symbol Editor 6-15
Editing Components 6-16
Editing a Component's General Properties 6-17
Editing a Component's Symbol 6-17
Copying a Component's Symbol 6-19
Copying a Multi-Section Component's Symbol 6-19
Creating and Editing a Component's Symbol with the Symbol Editor 6-19
Symbol Editor Spreadsheet View 6-21
Working with the Symbol Editor 6-23
Enter Text Dialog Box 6-27
In-Place Edit Mode 6-28
Symbol Editor Menus 6-28
Symbol Editor Toolbars 6-33
Editing a Component's Model 6-39
Adding a Model to a Component 6-41
Component List Dialog Box 6-42
Creating a SPICE Model for a Component 6-42
Creating a Model Using a Model Maker 6-42
Creating a Primitive Model 6-42
Creating a SPICE Subcircuit 6-44

Loading an Existing Model 6-45
Modify a Model's Data 6-46
Copying the Model of One Component to Another 6-46
Analog Behavioral Modeling and Controlled Source Syntax 6-47
Accessing net voltages and branch currents in ABM expressions 6-47
Editing a Component Pin Model 6-48
Editing a Component's Footprint 6-49
Select a Footprint dialog box 6-49
Filter dialog box 6-51
Add a Footprint dialog box 6-52
Advanced Pin Mapping Dialog 6-53
Editing a Component's Electronic Parameters 6-56
Editing User Fields 6-56
Creating a Component Model Using the Model Makers 6-57
AC Motor 6-57
BJT Model Maker 6-58
Converters 6-69
Boost Converter 6-70
Buck Boost Converter 6-71
Buck Converter 6-71
Cuk Converter 6-71
Diode Model Maker 6-72
Transformers 6-75
Ideal Transformer (Multiple Winding) 6-75
Linear Transformer (Multiple Winding) 6-76
Linear Transformer with Neutral Terminal 6-76
Two Winding Linear Transformer 6-76
Non-linear Transformer (Multiple Winding) 6-77
MOSFET (Field Effect Transistor) Model Maker 6-78
Operational Amplifier Model Maker 6-84
Silicon Controlled Rectifier Model Maker 6-87
Zener Model Maker 6-91
Creating a Model Using Code Modeling 6-94
What is Code Modeling? 6-94
Creating a Code Model 6-95
The Interface File (Ifspec.ifs) 6-99
Name Table 6-99
Port Table 6-101
Parameter Table 6-103
Example Interface File 6-104
The Implementation File (Cfunc.mod) 6-106
Implementation File C Macros 6-106
Example Implementation File 6-113

Chapter 7
Multisim SPICE Reference

Documentation Conventions 7-1
General Purpose Syntax 7-2
Primitive Device Declarations 7-2
SPICE Subcircuits 7-4
Circuit Parameters 7-7
Number Format 7-8
Comments and Line Continuation 7-9
Mathematical Expressions 7-10
Introduction 7-10
Supported Mathematical Functions, Operators and Constants 7-11
User-Defined Functions 7-17
Analog Devices 7-18
Resistor 7-19
Capacitor 7-20
Inductor 7-22
Coupled (Mutual) Inductor 7-23
Diode 7-24
Lossless Transmission Line 7-26
Lossy Transmission Line 7-27
Uniform R.C. Line (Lumped-approximation R.C. line) 7-28
JFET 7-29
MESFET 7-31
Voltage-Controlled Switch 7-34
Smooth Transition Voltage-Controlled Switch 7-35
Current-Controlled Switch 7-36
BJT 7-37
MOSFET 7-42
Independent Voltage Source 7-47
Pulse Source 7-50
Sinusoidal Source 7-51
Single Frequency FM Source 7-51
Exponential Source 7-52
Piecewise Linear Source 7-52
Piecewise Linear File Source 7-53
Modulation Source 7-53
AM Modulation Source 7-54
FM Modulation Source 7-54
FSK Modulation Source 7-55
BST Modulation Source 7-55
SWP Modulation Source 7-56
Noise Source 7-57

XARB Source 7-57
Independent Current Source 7-58
Arbitrary Source 7-60
Voltage Controlled Voltage Source 7-61
Polynomial Specifications 7-62
Current Controlled Voltage Source 7-63
Voltage Controlled Current Source 7-64
Current Controlled Current Source 7-65
Temperature Parameters 7-66
Global Temperature Parameters 7-66
Instance/Model Temperature Parameters 7-67
XSPICE Syntax Reference 7-68
XSPICE Code Model 7-68
Unusual Forms of Device Syntax 7-69
Compatibility Modes 7-70

Chapter 8
Simulation

Introduction to Simulation 8-1
Using Multisim Simulation 8-2
Interactive Components 8-3
Component Tolerances in Multisim 8-3
Start/Stop/Pause Simulation 8-4
Simulation Running Indicator 8-5
Simulation Speed 8-5
Circuit Consistency Check 8-6
Simulation from Netlist Without Schematic 8-6
Multisim SPICE Simulation: Technical Detail 8-6
Circuit Simulation Mechanism 8-7
Four Stages of Circuit Simulation 8-7
Equation Formulation 8-9
Equation Solution 8-10
User Setting: Maximum Integration Order 8-10
DC Convergence Assistance Algorithms 8-11
Digital Simulation 8-13
RF Simulation 8-14
MultiVHDL 8-15
Circuit Wizards 8-15
555 Timer Wizard 8-16
Filter Wizard 8-18
Common Emitter BJT Amplifier Wizard 8-20
Op-amp Wizard 8-21

Netlist and Simulation Errors 8-22
Netlist Errors and Warnings 8-22
Simulation Errors 8-24
Simulation Error Log/Audit Trail 8-24
Simulation Error Help 8-26
Convergence Assistant 8-26
Saving/Loading Simulation Profiles 8-27
Saving a Simulation Profile 8-28
Loading a Simulation Profile 8-28

Chapter 9
Instruments

Introduction to the Multisim Instruments 9-1
Saving Simulation Data with Instruments 9-2
Adding an Instrument to a Circuit 9-2
Using the Instrument 9-3
Working with Multiple Instruments 9-4
Saving Instrument Data 9-4
Printing Instruments 9-5
Interactive Simulation Settings 9-5
Troubleshooting Simulation Errors 9-6
Multimeter 9-7
Multimeter Settings 9-8
Function Generator 9-11
Function Generator Settings 9-12
Wattmeter 9-13
Connecting the Wattmeter 9-14
Oscilloscope 9-15
Oscilloscope Settings 9-16
Viewing Oscilloscope Results 9-19
Bode Plotter 9-19
Bode Plotter Settings 9-21
Viewing Bode Plotter Results 9-22
Word Generator 9-23
Word Generator Settings 9-24
Logic Analyzer 9-26
Logic Analyzer Settings 9-28
Logic Converter 9-31
Logic Converter Settings 9-32
Distortion Analyzer 9-34
Distortion Analyzer Settings 9-35
Spectrum Analyzer 9-36
Network Analyzer 9-36

Measurement Probe 9-37
Measurement Probe Settings 9-37
Dynamic Probe Settings 9-37
Static Probe Settings 9-38
Using the Dynamic Measurement Probe 9-40
Using the Static Measurement Probe 9-41
Four-Channel Oscilloscope 9-43
Four-channel Oscilloscope Settings 9-44
Viewing Four-channel Oscilloscope Results 9-47
Connecting the Four-channel Oscilloscope 9-50
Frequency Counter 9-50
Using the Frequency Counter 9-51
IV Analyzer 9-53
Using the IV Analyzer 9-54
Simulate Parameters Dialog Box 9-57
Reviewing IV Analyzer Data 9-61
Agilent Simulated Instruments 9-63
Agilent Simulated Function Generator 9-63
Supported Features 9-63
Features Not Supported 9-64
Using the Agilent Function Generator 9-64
Agilent Simulated Multimeter 9-65
Supported Features 9-65
Features Not Supported 9-65
Using the Agilent Multimeter 9-66
Agilent Simulated Oscilloscope 9-66
Supported Features 9-66
Features Not Supported 9-68
Using the Agilent Oscilloscope 9-68
Tektronix Simulated Instruments 9-69
Tektronix Simulated Oscilloscope 9-69
Supported Features 9-69
Features Not Supported 9-71
Using the Tektronix Oscilloscope 9-72
Voltmeter 9-72
Using the Voltmeter 9-72
Ammeter 9-73
Using the Ammeter 9-74
Current Probe 9-74
Current Probe Properties 9-75
LabVIEW Instruments 9-78
System Requirements 9-79
Sample LabVIEW Instruments 9-80
Microphone 9-80
Speaker 9-81

Signal Analyzer 9-81
Signal Generator 9-82
Streaming Signal Generator 9-82
LabVIEW Instruments Toolbar 9-83
Creating a LabVIEW Instrument 9-83
Building a LabVIEW Instrument 9-86
Installing a LabVIEW Instrument 9-86
Guidelines for Successfully Creating a LabVIEW Instrument 9-86

Chapter 10
Analyses

Introduction to Multisim Analyses 10-1
Viewing the Analysis Results: Grapher 10-2
Working with Pages on the Grapher 10-6
Working with Graphs 10-7
Grids and Legends 10-7
Cursors 10-8
Cursor Pop-up Menu 10-10
Zoom and Restore 10-11
Title 10-11
Axes 10-12
Traces 10-13
Merging Traces 10-14
Select Pages dialog box 10-14
Graph Pop-up Menu 10-15
Viewing Charts 10-15
Cut, Copy and Paste 10-16
Opening Files 10-17
Saving Files 10-17
Print and Print Preview 10-19
Adding Traces from the Latest Simulation Results 10-20
Working with Analyses 10-21
General Instructions 10-22
Analysis Parameters Tab 10-22
Output Tab 10-23
Choosing How to Handle Output Variables 10-23
Filtering the Variable Lists 10-24
Adding Parameters to the Variable List 10-24
Selecting Variables to Save 10-25
Adding Analysis Expressions 10-26
Analysis Options Tab 10-28
Summary Tab 10-29
Incomplete Analyses 10-29

DC Operating Point Analysis 10-30
Setting up and Running DC Operating Point Analysis 10-30
Setting DC Operating Point Analysis Parameters 10-30
Sample Circuit 10-31
Troubleshooting DC Operating Point Analysis Failures 10-32
Circuit Failure Example 10-33
Trouble-shooting Techniques 10-34
AC Analysis 10-35
Setting AC Analysis Frequency Parameters 10-36
Transient Analysis 10-37
Setting Transient Analysis Parameters 10-38
Troubleshooting Transient Analysis Failures 10-40
Fourier Analysis 10-40
Setting Fourier Analysis Parameters 10-41
Noise Analysis 10-44
Setting Noise Analysis Parameters 10-45
Noise Analysis Example 10-48
Distortion Analysis 10-50
Multisim Approach 10-51
Preparing the Circuit for Distortion Analysis 10-51
Understanding the Distortion Analysis Options 10-52
Distortion Analysis for Harmonic Distortion 10-53
Distortion Analysis for Intermodulation Distortion 10-55
DC Sweep Analysis 10-57
Setting DC Sweep Analysis Parameters 10-57
DC Sweep Analysis Examples 10-59
DC and AC Sensitivity Analyses 10-62
Sensitivity Analysis Parameters 10-62
Setting Up and Running Sensitivity Analysis 10-64
Parameter Sweep Analysis 10-67
Setting Parameter Sweep Analysis Parameters 10-67
Parameter Sweep Analysis – Example 10-70
Temperature Sweep Analysis 10-74
Setting Temperature Sweep Analysis Parameters 10-75
Transfer Function Analysis 10-76
Setting Transfer Function Analysis Parameters 10-77
Examples of Transfer Function Analysis 10-79
Worst Case Analysis 10-82
Setting Tolerance Parameters 10-83
Setting Worst Case Analysis Parameters 10-84
Worst Case Analysis Example 10-85
Pole Zero Analysis 10-87
Multisim Approach 10-90
Setting Pole Zero Analysis Parameters 10-91

Running Pole Zero Analysis 10-92
Monte Carlo Analysis 10-93
Uniform Distribution 10-94
Gaussian Distribution 10-94
Setting Up and Running Monte Carlo Analysis 10-96
Entering a Component Tolerance 10-96
Specifying Monte Carlo Analysis Parameters 10-97
Monte Carlo Analysis Example 10-99
Setting up the Sample Monte Carlo Analysis 10-100
Simulation Results 10-104
Trace Width Analysis 10-107
Multisim Approach 10-109
Sample Trace Width Analysis 10-111
RF Analyses 10-112
Nested Sweep Analyses 10-113
Batched Analyses 10-113
User Defined Analyses 10-115
Creating and Simulating a SPICE Netlist 10-115
Importing the SPICE netlist into Multisim 10-116
Plotting Two Nodes Using the Tran Statement 10-117
Custom Analysis Options Dialog Box 10-117
Global Tab 10-118
DC Tab 10-120
Transient Tab 10-122
Device Tab 10-124
Advanced Tab 10-125

Chapter 11
Postprocessor

Introduction to the Postprocessor 11-1
Using the Postprocessor 11-2
Basic Steps 11-2
Using the Default Analysis 11-7
Creating Multiple Traces 11-8
Working with Pages, Traces, Graphs and Charts 11-8
Postprocessor Variables 11-9
Available Postprocessor Functions 11-9

Chapter 12
Reports

Bill of Materials 12-1
Using the BOM Report 12-3
Component Detail Report 12-5
Using the Component Detail Report 12-5
Netlist Report 12-6
Using the Netlist Report 12-6
Schematic Statistics Report 12-7
Using the Schematic Statistics Report 12-8
Spare Gates Report 12-9
Using the Spare Gates Report 12-9
Model Data Report 12-10
Using the Model Data Report 12-10
Cross Reference Report 12-11
Using the Cross Reference Report 12-11
Variants Filter Dialog Box 12-12

Chapter 13
Transfer/Communication

Introduction to Transfer/Communication 13-1
Exporting to PCB layout 13-1
Transferring from Multisim to Ultiboard for PCB Layout 13-2
Transfering to Other PCB Layout Packages 13-3
Forward Annotation 13-4
Back Annotation 13-5
Exporting Simulation Results 13-5
Exporting to MathCAD 13-5
Exporting to Excel 13-6
Saving to a Measurement File 13-7
Exporting a Netlist 13-8
Importing Files with Other Formats 13-8
Opening an Electronics Workbench 5 File 13-9
Opening an Ulticap File 13-9
Importing a SPICE or Cadence® PSpice® Netlist 13-10

Chapter 14
RF

Introduction to the Multisim RF Module 14-1
RF Components 14-2
Multisim's RF Components 14-3
Theoretical Explanation of the RF Models 14-3
Striplines/Microstrips/Waveguides 14-3
RF Resistors 14-5
RF Capacitors 14-6
RF Inductors 14-7
Devices 14-8
RF Instruments 14-9
Spectrum Analyzer 14-9
Using Multisim's Spectrum Analyzer 14-10
Frequency Range 14-11
Frequency Spans 14-11
Frequency Analysis 14-11
Amplitude Range 14-12
Reference Level 14-13
Frequency Resolution 14-13
Examples 14-14
Network Analyzer 14-17
Using the Network Analyzer 14-17
Marker Controls 14-18
Trace Controls 14-19
Format Controls 14-19
Data Controls 14-19
Mode Controls 14-20
RF Analyses 14-20
RF Characterizer Analysis 14-20
Matching Network Analysis 14-23
Noise Figure Analysis 14-28
Multisim Approach 14-31
Noise Figure Analysis Tabs 14-31
Sample Noise Figure Analysis 14-32
RF Model Makers 14-33
Waveguide 14-34
Microstrip Line 14-35
Open End Microstrip Line 14-36
RF Spiral Inductor 14-36
Strip Line Model 14-37
Stripline Bend 14-38

Lossy Line 14-39
Interdigital Capacitor 14-40
Tutorial: Designing RF Circuits 14-41
Selecting Type of RF Amplifier 14-42
Selecting an RF Transistor 14-42
Selecting a DC Operating Point 14-43
Selecting the Biasing Network 14-43
Selecting an Operating Frequency Point 14-45
Analyzing the RF Network 14-45

Chapter 15
Multisim MCU

Introduction to Multisim MCU 15-1
Multisim MCU Basics 15-1
Debugging Tools 15-2
MCU Source Code Editor 15-2
MCU Memory View 15-3
Advanced Features 15-3
Peripheral Devices 15-3
Multisim MCU File Management 15-4
MCU Design Overview 15-4
MCU Wizard 15-5
Adding and Removing Projects, Folders and Files 15-7
Multisim MCU Code Manager 15-8
MCU Code Manager Overview 15-8
Adding and Removing Projects, Folders and Files in the MCU Code Manager 15-10
MCU Project Build Settings 15-11
Documentation for Supported Tools 15-12
8051 Workspace Build Settings Example 15-13
Loading an External Hex File Project 15-19
Building an MCU Workspace 15-20
Errors and Warnings 15-21
Simulation of Machine Code File 15-21
Multisim MCU Source Code Editor 15-24
Opening a Source Code File 15-24
Building Source Code Files 15-24
Source Code View 15-25
MCU Debugging Features 15-26
Definitions 15-26
Opening a Debug View 15-26
Debug Window Settings 15-27
MCU Toolbar Buttons 15-28

Simulation Markers 15-29
Breakpoints 15-29
Stepping and Breaking 15-31
Memory View 15-32

Chapter 16
Automation API

Appendix A
Menus and Commands

File Menu..... A-1
Edit Menu A-4
View Menu A-6
Place Menu A-9
MCU Menu A-11
Simulate Menu A-11
Transfer Menu A-18
Tools Menu A-19
Reports Menu A-22
Options Menu A-23
Window Menu A-23
Help Menu..... A-24

Appendix B
Archiving Data

Backing up the Corporate Database B-1
Backing up the User Database and Configuration Files B-2

Appendix C
Technical Support and Professional Services

Index

1

User Interface

This chapter explains the basic operation of the Multisim user interface.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Introduction to the Multisim Interface

Multisim is the schematic capture and simulation application of National Instruments Circuit Design Suite, a suite of EDA (Electronics Design Automation) tools that assists you in carrying out the major steps in the circuit design flow. Multisim is designed for schematic entry, simulation, and feeding to downstage steps, such as PCB layout.

Note For more information on the Multisim interface, and information on the other components of NI Circuit Design Suite, refer to the *Getting Started with NI Circuit Design Suite* manual.

Toolbars

The toolbars listed below are available in Multisim:

- *Standard Toolbar*
- *Main Toolbar*
- *Simulation Toolbar*
- *View Toolbar*
- *Components Toolbar*
- *Virtual Toolbar*
- *Graphic Annotation Toolbar*
- *Instruments Toolbar*

Note If the above toolbars are not visible, select **View»Toolbars»<toolbar name>**.

Standard Toolbar

The **Standard** toolbar contains buttons for commonly performed functions. Its buttons are described below:

Button	Description
	New button. Creates a new circuit file.
	Open button. Opens an existing circuit file.
	Open Sample button. Opens a folder containing sample and getting started files.
	Save button. Saves the active circuit.
	Print Circuit button. Prints the active circuit.
	Print Preview button. Previews the circuit as it will be printed.
	Cut button. Removes the selected elements and places them on the Windows clipboard.
	Copy button. Copies the selected elements and places them on the Windows clipboard.
	Paste button. Inserts the contents of the Windows clipboard at the cursor location.
	Undo button. Undoes the most recently performed action.
	Redo button. Redoes the most recently performed undo.

Main Toolbar

The buttons in the **Main** toolbar are described below:

Button	Description
	Toggle Design Toolbox button. Switches the **Design Toolbox** on and off. Refer to the *Design Toolbox* section for more information.
	Toggle Spreadsheet View button. Switches the **Spreadsheet View** on and off. Refer to the *Spreadsheet View* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.
	Database Manager button. Launches the **Database Manager** dialog box. Refer to the *Editing Components* section of Chapter 6, *Component Editing*, for more information.
	Create Component button. Launches the **Component Wizard**. Refer to the *Creating Components with the Component Wizard* section of Chapter 6, *Component Editing*, for more information.
	Grapher/Analyses button. Displays the grapher. Refer to the *Viewing the Analysis Results: Grapher* section of Chapter 10, *Analyses*, for more information. Also displays a list of available analyses. Refer to the *Working with Analyses* section of Chapter 10, *Analyses*, for more information.
	Postprocessor button. Displays the **Postprocessor** dialog box. Refer to the *Using the Postprocessor* section of Chapter 11, *Postprocessor*, for more information.
	Electrical Rules Checking button. Checks that the electrical rules established for the wiring of the circuit have been followed. Refer to the *Electrical Rules Checking* section of Chapter 3, *User Interface*, for more information.
	Capture Screen Area button. Refer to the *Capturing Screen Area* section of Chapter 2, *Schematic Capture - Basics*, for more details.

Button	Description
	Go to Parent Sheet button. Displays the top level sheet in a hierarchical design. Refer to the *Hierarchical Design* and *Viewing a Parent Sheet* sections of Chapter 4, *Working with Larger Designs*, for more details.
	Back Annotate from Ultiboard button. Refer to the *Back Annotation* section of Chapter 13, *Transfer/Communication*, for more information.
	Forward Annotate button. Refer to the *Forward Annotation* section of Chapter 13, *Transfer/Communication*, for more information.
—	**In Use List** (not shown). Click on the down-arrow to display a list of the active circuit's components. Refer to the *Using the In Use List* section of Chapter 2, *Schematic Capture - Basics*, for more details.
	Help button. Launches the help file.

Simulation Toolbar

The **Simulation** toolbar contains buttons used during simulation.

Button	Description
	Run/resume simulation button. Starts/resumes simulation of the active circuit. Refer to the *Start/Stop/Pause Simulation* section of Chapter 8, *Simulation*, for more information.
	Pause simulation button. Pauses simulation. Refer to the *Start/Stop/Pause Simulation* section of Chapter 8, *Simulation*, for more information.
	Stop simulation button. Stops the simulation. Refer to the *Start/Stop/Pause Simulation* section of Chapter 8, *Simulation*, for more information.
	Pause at Next MCU Instruction Boundary button. Refer to the *Stepping and Breaking* section for more information.

Button	Description
	Step Into button. Refer to the *Stepping and Breaking* section for more information.
	Step Over button. Refer to the *Stepping and Breaking* section for more information.
	Step Out button. Refer to the *Stepping and Breaking* section for more information.
	Run to Cursor button. Refer to the *Stepping and Breaking* section for more information.
	Toggle Breakpoint button. Refer to the *Breakpoints* section for more information.
	Remove All Breakpoints button. Refer to the *Breakpoints* section for more information.

View Toolbar

The buttons in the **View** toolbar are described below:

Button	Description
	Toggle Full Screen button. Displays only the workspace, with no toolbars or menu items.
	Increase Zoom button. Magnifies the active circuit.
	Decrease Zoom button. Decreases the magnification of the active circuit.
	Zoom to Selected Area button. Drag the cursor to select an area on the workspace to magnify.
	Zoom Fit to Page button. Shows the entire circuit in the workspace.

Components Toolbar

The buttons in the **Components** toolbar are described below. Each button will launch the place component browser (**Select a Component** browser) with the group specified on the button pre-selected. Refer to the *Using the Place Component Browser* section for more information.

Button	Description
	Place Source button. Selects the **Source** components group in the browser.
	Place Basic button. Selects the **Basic** components group in the browser.
	Place Diode button. Selects the **Diode** components group in the browser.
	Place Transistor button. Selects the **Transistor** components group in the browser.
	Place Analog button. Selects the **Analog** components group in the browser.
	Place TTL button. Selects the **TTL** components group in the browser.
	Place CMOS button. Selects the **CMOS** component group in the browser.
	Place Miscellaneous Digital button. Selects the **Miscellaneous Digital** component group in the browser.
	Place Mixed button. Selects the **Mixed** component group in the browser.
	Place Power Components button. Selects the **Power** component group in the browser.
	Place Indicator button. Selects the **Indicator** component group in the browser.

Button	Description
	Place Miscellaneous button. Selects the **Miscellaneous** component group in the browser.
	Place Advanced Peripherals button. Selects the **Advanced Peripherals** component group in the browser.
	Place RF button. Selects the **RF** component group in the browser.
	Place Electromechanical button. Selects the Electromechanical component group in the browser.
	Place MCU button. Selects the **MCU** component group in the browser.
	Place Hierarchical Block button. Opens a file to be embedded as a hierarchical block. Refer to the *Hierarchical Design* section of Chapter 4, *Working with Larger Designs*, for more information.
	Place Bus button. Places a bus with segments created as you click on the workspace. Refer to the *Buses* section of Chapter 4, *Working with Larger Designs*, for more information.

Virtual Toolbar

Use the **Virtual** toolbar to place virtual components on your workspace. Refer to the *Placing Virtual Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Graphic Annotation Toolbar

The buttons in the **Graphic Annotation** toolbar are described below. Refer to the *Graphic Annotation* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Button	Description
	Picture button. Click on this button to place a picture on the workspace.
	Polygon button. Click on this button to draw a polygon.
	Arc button. Click on this button to draw an arc.
	Ellipse button. Click on this button to draw an ellipse. .
	Rectangle button. Click on this button to draw a rectangle.
	Multiline button. Click on this button to draw a multiline.
	Line button. Click on this button to draw a line.
	Place Text button. Places a text frame on your workspace into which you can enter miscellaneous text. Refer to the *Adding Miscellaneous Text* section of Chapter 2, *Schematic Capture - Basics*, for more information.
	Place Comment button. Click on this button to place a comment on the workspace. Refer to the *Adding a Comment* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Instruments Toolbar

The buttons in the **Instruments** toolbar are described below. In each case, the button places a specific instrument on the workspace.

Some versions of Multisim do not include all of the instruments described below.

Button	Description
	Multimeter button. Places a multimeter on the workspace. Refer to the *Multimeter* section of Chapter 9, *Instruments*, for more information.
	Function Generator button. Places a function generator on the workspace. Refer to the *Function Generator* section of Chapter 9, *Instruments*, for more information.
	Wattmeter button. Places a wattmeter on the workspace. Refer to the *Wattmeter* section of Chapter 9, *Instruments*, for more information.
	Oscilloscope button. Places an oscilloscope on the workspace. Refer to the *Oscilloscope* section of Chapter 9, *Instruments*, for more information.
	Four Channel Oscilloscope button. Places a four-channel oscilloscope on the workspace. Refer to the *Four-Channel Oscilloscope* section of Chapter 9, *Instruments*, for more information.
	Bode Plotter button. Places a Bode plotter on the workspace. Refer to the *Bode Plotter* section of Chapter 9, *Instruments*, for more information.
	Frequency Counter button. Places a frequency counter on the workspace. Refer to the *Frequency Counter* section of Chapter 9, *Instruments*, for more information.
	Word Generator button. Places a word generator on the workspace. Refer to the *Word Generator* section of Chapter 9, *Instruments*, for more information.
	Logic Analyzer button. Places a logic analyzer on the workspace. Refer to the *Logic Analyzer* section of Chapter 9, *Instruments*, for more information.
	Logic Converter button. Places a logic converter on the workspace. Refer to the *Logic Converter* section of Chapter 9, *Instruments*, for more information.
	IV-Analysis button. Places an IV Analyser on the workspace. Refer to the *IV Analyzer* section of Chapter 9, *Instruments*, for more information.

Button	Description
	Distortion Analyzer button. Places a distortion analyzer on the workspace. Refer to the *Distortion Analyzer* section of Chapter 9, *Instruments*, for more information.
	Spectrum Analyzer button. Places a spectrum analyzer on the workspace. Refer to the *Spectrum Analyzer* section of Chapter 9, *Instruments*, for more information.
	Network Analyzer button. Places a network analyzer on the workspace. Refer to the *Network Analyzer* section of Chapter 9, *Instruments*, for more information.
	Agilent Function Generator button. Places an Agilent function generator on the workspace. Refer to the *Agilent Simulated Function Generator* section of Chapter 9, *Instruments*, for more information.
	Agilent Multimeter button. Places an Agilent multimeter on the workspace. Refer to the *Agilent Simulated Multimeter* section of Chapter 9, *Instruments*, for more details.
	Agilent Oscilloscope button. Places an Agilent oscilloscope on the workspace. Refer to the *Agilent Simulated Oscilloscope* section of Chapter 9, *Instruments*, for more information.
	Tektronix Oscilloscope button. Places a Tektronix oscilloscope on the workspace. Refer to the *Tektronix Simulated Oscilloscope* section of Chapter 9, *Instruments*, for more information.
	Current Probe button. Places a current probe on the workspace. Refer to the *Current Probe* section of Chapter 9, *Instruments*, for more information.

Button	Description
	LabVIEW Instruments button. Places a LabVIEW instrument on the workspace. Refer to the *LabVIEW Instruments* section of Chapter 9, *Instruments*, for more information.
	Measurement Probe button. Attaches a probe to the mouse pointer that measures voltage, current and frequency on any wire on your schematic. Can be placed before simulation (static probe), or during simulation (dynamic probe). Use the arrow to place static probes. Refer to the *Measurement Probe* section of Chapter 9, *Instruments*, for more information.

Locking Toolbars

You can lock toolbars in place to prevent them from being inadvertently moved.

Once you have placed them as you like, you can lock all *docked* toolbars in place in different ways:

- Select **Options»Lock Toolbars**.
- Enable **Lock all docked toolbars** from the **Toolbars** tab of the **Customize** dialog box. Refer to the *Toolbars tab* section for more information.
- Right-click in any toolbar and select **Lock Toolbars** from the pop-up menu that appears.

Note If the **Lock Toolbars** option is enabled, floating toolbars will be locked in place if you move them to a docked position.

Tip To drag a docked toolbar to a different location when **Lock Toolbars** is enabled, hold down <Alt> and drag the toolbar with your mouse.

When **Lock Toolbars** is *disabled*, the drag bars on all docked toolbars display, as shown in the figure below (1). Grab the toolbar by the drag bar to move it.

1 Drag Bar

Note If the **Lock Toolbars** option is enabled, the drag bars do not appear.

Toolbar Behavior When Multisim Size is Reduced

If you reduce the size of Multisim, some of the buttons in larger toolbars may become hidden. If this happens, the **More Buttons** icon appears as shown in the figure below (1).

1 More Buttons Icon

Click the **More Buttons** icon to display the hidden buttons.

Note Text boxes and drop-down lists do not display when Multisim's size is reduced. To show these, make Multisim large enough to show all toolbar buttons without the **More Buttons** icon.

Tip You can make your own toolbars and add or delete buttons from existing toolbars to get everything to fit. Refer to the *Toolbars tab* section for more information.

Pop-up Menus

In addition to the standard menus at the top of the screen (for example, **File**, **Edit**), there are also a number of context-sensitive pop-up menus available.

Pop-up From Circuit Window, with no Component Selected

When you right-click on the circuit window with no component selected, a pop-up menu of appropriate commands appears. These commands are:

- **Place Component**—Lets you browse the databases (**Master**, **Corporate** and **User**) for components to be placed. Refer to the *Using the Place Component Browser* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Schematic»Component**—Lets you browse the databases (**Master**, **Corporate** and **User**) for components to be placed. Refer to

the *Using the Place Component Browser* section of Chapter 2, *Schematic Capture - Basics*, for more information.

- **Place Schematic»Junction**—Places a connector. Refer to the *Manually Adding a Junction* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Schematic»Wire**—Use to place a wire on the workspace. Refer to the *Placing Wires Directly Onto Workspace* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Schematic»Bus**—Places a bus with segments created as you click. Refer to the *Buses* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Place Schematic»HB/SC Connector**—Adds connectors to a circuit for use in a hierarchical block or a subcircuit.
- **Place Schematic»Off-Page Connector**—Places an off-page connector on your workspace. Refer to the *Flat Multi-Sheet Design* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Place Schematic»Bus HB/SC Connector**—Adds bus connectors to a circuit for use in a hierarchical block or a subcircuit.
- **Place Schematic»Bus Off-Page Connector**—Places an off-page connector for use with buses only on your workspace.
- **Place Schematic»Hierarchical Block from File**—Opens a file to be embedded as a hierarchical block. Refer to the *Placing a HB from an Existing File* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Place Schematic»New Hierarchical Block**—Displays the **Hierarchical Block Properties** dialog box. Refer to the *Adding a Hierarchical Block* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Place Schematic»New Subcircuit**—Places a new subcircuit on the workspace. Refer to the *Adding a Subcircuit* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Place Schematic»Replace by Subcircuit**—Replaces the selection by a subcircuit.
- **Place Schematic»Multi-Page**—Opens a new page. Refer to the *Flat Multi-Sheet Design* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Place Schematic»Merge Bus**—Merges selected buses.

- **Place Schematic»Bus Vector Connect**—Use to place numerous connections from a multi-pinned device, such as an IC, to a bus. Refer to the *Bus Vector Connect* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Place Graphic»Text**—Places text on the circuit. Refer to the *Adding Miscellaneous Text* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Graphic»Line**—Places a straight line on your workspace. Refer to the *Graphic Annotation* section for more information.
- **Place Graphic»Multiline**—Places a multiline on your workspace. Refer to the *Graphic Annotation* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Graphic»Rectangle**—Places a rectangle on your workspace. Refer to the *Graphic Annotation* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Graphic»Ellipse**—Places an ellipse on your workspace. Refer to the *Graphic Annotation* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Graphic»Arc**—Places an arc on your workspace. Refer to the *Graphic Annotation* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Graphic»Polygon**—Places a polygon on your workspace. Refer to the *Graphic Annotation* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Graphic»Picture**—Places a picture on your workspace. Refer to the *Graphic Annotation* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Place Comment**—Use to “pin” a comment to the workspace, or directly to a component. Refer to the *Adding a Comment* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Cut**—Removes the selected item from the circuit and places it on the clipboard.
- **Copy**—Copies the selected item from the circuit to the clipboard.
- **Paste**—Pastes the current contents of the clipboard onto the circuit.
- **Delete**—Deletes the selection from the workspace.
- **Select All**—Selects all elements on the workspace.
- **Toggle NC Marker**—Places an NC (no connection) marker on a component’s pin. Refer to the *Marking Pins for No Connection* section of Chapter 2, *Schematic Capture - Basics*, for more information.

- **Clear ERC Markers**—Clears existing ERC (electrical rules check) markers from the workspace. Refer to the *ERC Options Tab* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.
- **Paste as Subcircuit**—Pastes the contents of the clipboard onto the workspace as a subcircuit.
- **Replace by Subcircuit**—Replaces the elements that you have selected with a subcircuit. Refer to the *Hierarchical Design* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Font**—Displays a dialog box where you set font information for the circuit. Refer to the *Sheet Properties - Font Tab* section for more information.
- **Properties**—Displays the **Sheet Properties** dialog box. Be sure that you have no elements on the workspace selected, or the properties for that element will appear instead. Refer to the *Using the Sheet Properties Dialog Box* section for more information.

Pop-up From a Selected Component or Instrument

When you right-click on a selected element (component or instrument), a pop-up menu of appropriate commands appears. These commands are:

- **Cut**—Removes the selected components, circuits or text, and places them on the clipboard.
- **Copy**—Copies the selected components, circuits or text, and places them on the clipboard. Refer to the *Copying a Placed Component* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Paste**—Places the contents of the clipboard on the workspace. The cursor shows a “ghosted” image of the item to be pasted. Click to indicate where the item is to be pasted.
- **Delete**—Deletes the selection from the workspace.
- **Flip Horizontal** —Flips the selection horizontally. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Flip Vertical**—Flips the selection vertically. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **90 Clockwise**—Rotates the selection 90 degrees clockwise. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.

- **90 CounterCW**—Rotates the selection 90 degrees counterclockwise. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Bus Vector Connect**—Displays the **Bus Vector Connect** dialog box. Refer to the *Bus Vector Connect* section for more information.
- **Replace by Hierarchical Block**—Replaces the elements that you have selected with a hierarchical block. Refer to the *Replacing Components with an HB* section for more information.
- **Replace by Subcircuit**—Replaces the elements that you have selected with a subcircuit. Refer to the *Hierarchical Design* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Replace Components**—Invokes the **Select a Component** browser from which you can select a new component. Refer to the *Replacing a Placed Component* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Save Component to DB**—Lets you save the selected component, including any changes you made to it, to the database. Refer to the *Saving Placed Components* section of Chapter 5, *Components*, for more information.
- **Edit Symbol/Title Block**—Depending on the selected item, launches either the **Symbol Editor** or the **Title Block Editor**.
- **Reverse Probe Direction**—Reverses the polarity of a selected Measurement Probe or Current Probe.
- **Change Color**—Displays a color palette where you can change the color of the selected element's lines.
- **Font**—Changes the font of various elements on the workspace from their default values. Refer to the *Sheet Properties - Font Tab* section for more information.
- **Properties**—If a component is selected, displays that component's properties dialog box. If an instrument is selected, opens the instrument face.

Pop-up From a Selected Wire

When you right-click on a selected wire in the circuit window, a pop-up menu of appropriate commands appears. These commands are:

- **Delete**—Deletes the selected wire from the workspace.
- **Change Color**—Changes the color of the selected wire from its default value.
- **Color Segment**—Changes the color of the selected wire *segment* from its default value.
- **Font**—Changes the font of various elements on the workspace from their default values. Refer to the *Sheet Properties - Font Tab* section for more information.
- **Properties**—Displays the **Net** dialog box. Refer to the *Modifying Net Names* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Pop-up From a Selected Text Block or Graphic

When you right-click on a selected text block or graphic element in the circuit window, a pop-up menu of appropriate commands appears. These commands are:

- **Delete**—Deletes the selected item.
- **Flip Horizontal**—Flips the selection horizontally. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Flip Vertical**—Flips the selection vertically. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **90 Clockwise**—Rotates the selection 90 degrees clockwise. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **90 CounterCW**—Rotates the selection 90 degrees counterclockwise. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Pen Color**—Changes the color of the selected item from its default value.
- **Pen Style**—Changes the style of the pen for a selected graphic. Is disabled if text is selected.

- **Fill Color**—Changes the color of the fill for selected rectangles, ellipses and polygons. Is disabled if any other graphic element, or text is selected.
- **Fill Type**—Changes the appearance of the fill for selected rectangles, ellipses and polygons. Is disabled if any other graphic element, or text is selected.
- **Arrow**—Places an arrow on selected lines and multilines. Is disabled if any other graphic element, or text is selected.
- **Order**—Use to **Bring to Front** or **Send to Back** the selected element.
- **Assign to Layer**—Places selected element on the selected layer. Deselect to remove an element from the assigned layer. Refer to the *Sheet Properties - Visibility Tab* section for more information.
- **Font**—Lets you select a font, font style and size for selected text.
- **Properties**—Is inactive for text blocks or graphics.

Pop-up From a Title Block

When you right-click on a selected title block in the circuit window, a pop-up menu of appropriate commands appears. These commands are:

- **Delete**—Deletes the selected title block.
- **Edit Symbol/Title Block**—Launches the **Title Block Editor**. Refer to the *Title Block Editor* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.
- **Change Color**—Displays a palette where you can change the selection's color.
- **Move to»Bottom Left**—Places the selected title block at the bottom left corner of the document.
- **Move to»Bottom Right**—Places the selected title block at the bottom right corner of the document.
- **Move to»Top Left**—Places the selected title block at the top left corner of the document.
- **Move to»Top Right**—Places the selected title block at the top right corner of the document.
- **Properties**—Lets you change the information that you see in the title block. Refer to the *Entering the Title Block Contents* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Pop-up from a Comment or Measurement Probe

When you right-click on a selected **Comment** or **Measurement Probe**, a pop-up menu of appropriate commands appears. These commands are:

- **Cut**—Removes the selected item and places it on the clipboard.
- **Copy**—Copies the selected item and places it on the clipboard.
- **Paste**—Places the contents of the clipboard to the workspace. The cursor shows a "ghosted" image of the item to be pasted. Click to indicate where the item is to be pasted.
- **Delete**—Deletes the selected item from the workspace.
- **Show Comment/Probe**—Displays the contents of the comment or placed probe.
- **Edit Comment**—Is active for a selected comment only. Use to enter text into the comment.
- **Reverse Probe Direction**—Is active for a selected probe only. Reverses the probe's polarity.
- **Font**—Changes the font of various elements on the workspace from their default values. Refer to the *Sheet Properties - Font Tab* section for more information.
- **Properties**—Depending on the element selected, displays either the **Comment Properties**, or the **Probe Properties** dialog box. Refer to the *Adding a Comment* sectionof Chapter 2, *Schematic Capture - Basics*, and the *Measurement Probe Settings* section of Chapter 9, *Instruments*, for more information.

Setting Schematic Capture Preferences

You can customize virtually any aspect of Multisim preferences, including the colors used in your circuit, page size, zoom level, auto-backup interval, symbol set (ANSI or DIN) and printer setup. Your customization settings are saved individually with each circuit file you use so you could, for example, have one color scheme for one circuit and another for a different circuit. You can also override the settings for individual instances (for example, change one particular component from red to orange) or for the entire circuit.

Customization is done through the:

- **Preferences** dialog box—Used to set up global preferences. These preferences can vary from computer to computer. Refer to the *Using the Preferences Dialog Box* sections for more information.

- **Sheet Properties** dialog box—Used to set up the preferences for the active sheet. These preferences are saved with the circuit files so that if the circuit is opened on another computer, it will use the same settings. Refer to the *Using the Sheet Properties Dialog Box* section for more information

Using the Preferences Dialog Box

This section explains general procedures for setting preferences. The following sections describe details of setting specific options.

Note Refer to the *Using the Sheet Properties Dialog Box* section for information on sheet preferences.

Complete the following steps to set your user preferences:

1. Choose **Options»Global Preferences**. The **Preferences** dialog box appears, with the following tabs:
   - **Paths**—Use to change the filepaths for the databases, configuration files, and other items.
   - **Save**—Use to set up security copy creation, auto-backup timing and saving simulation data with instruments.
   - **Parts**—Use to set up component placement mode and the symbol standard (ANSI or DIN).
   - **General**—Use to set up selection rectangle behavior, mouse wheel behavior, autowiring behavior, and language.
   - **Simulation**—Use to set up netlist error behavior, default graph background color, and positive phase shift direction.
2. Select the desired tab.
3. Set the desired options. The specific options and settings available in the tabs are described in the following sections.
4. Click **OK** to save your changes.

Preferences - Paths Tab

The Multisim installation puts specific files in specific locations. If necessary you can point to a new location to find, for example, database files.

Tip Click on an item in the **Paths** tab to show its description.

Complete the following steps to change file locations:

1. Choose **Options»Global Preferences**. The **Preferences** dialog box appears.
2. Select the **Paths** tab.
3. Change the **General** settings as desired:
   - **Circuit Default Path**—This is where all new files are saved, unless you manually navigate to a new location when saving.
   - **User Button Images Path**—This is where you store any user-created button graphics.
4. Change the **User Settings** as desired:
   - **Configuration File**—Contains the user interface settings.
   - **New User Configuration File**—Click to create a new user configuration file and select an item from the drop-down list.
5. Change the settings in the **Database Files** area as desired:
   - **Master Database**—The location of the **Master Database**.
   - **Corporate Database**—The location of the **Corporate Database**.
   - **User Database**—The location of the **User Database**.

   Refer to the *Structure of the Component Database* section of Chapter 5, *Components*, for information on Multisim's databases.
6. Change the **Miscellaneous** settings as desired:
   - **User LabVIEW Instruments Path**—The location where user-created LabVIEW instruments are found. Refer to the *LabVIEW Instruments* section of Chapter 9, *Instruments*, for information on these.
   - **Code Models Path**—The location where user-created code models are found.

Preferences - Save Tab

The options in this tab let you set up security copy creation, auto-backup timing and saving simulation data with instruments.

Complete the following steps to set up save preferences:

1. Choose **Options»Global Preferences** and click on the **Save** tab that appears in the **Preferences** dialog box.
2. Set the following as desired:
   - **Create a Security Copy**—Enable to save a security copy when you do a save.

- **Auto-backup**—Enable to create a recovery file that is saved at the interval that you specify in the **Auto-backup interval** field.
- **Save simulation data with instruments**—Enable to save data displayed on instruments with the circuit file. Enter the maximum size of the simulation data in the **Maximum size** field.
- **Save .TXT files as plain text (not Unicode)**—When enabled, text files saved from reports will be in plain text file format, not Unicode text file format.

Preferences - Parts Tab

The options in this tab determine the component placement mode and the symbol standard (ANSI or DIN).

Complete the following steps to set up parts preferences:

1. Choose **Options»Global Preferences** and click on the **Parts** tab that appears in the **Preferences** dialog box.
2. In the **Place Component Mode** box:
   - **Return to Component Browser after placement** checkbox—Enable to return to the **Select a Component** dialog box after you place a component on the workspace. This is useful when you have many parts to place. Press <Esc> when finished part placement.
   - **Place single component** radio button—Select to place one selected component at a time.
   - **Continuous placement for multi-section part only** radio button—Select to place multiple sections of a multi-section component. For example, 7400N has four NAND gates, so using this option means each time you place a 7400N you place a different one of its NAND gates. Refer to the *Using the Place Component Browser* section of Chapter 2, *Schematic Capture - Basics*, for an example of multi-section part placement.
   - **Continuous placement** radio button—Select to place several components of the same type by continuing to click on the workspace after each component is placed. Press <Esc> to end continuous placement.
3. In the **Symbol Standard** box:
   - Select the symbol set to use for components (**ANSI** or **DIN**). The graphic changes to represent the selected symbol set. Refer to the *Creating and Editing a Component's Symbol with the Symbol Editor* section of Chapter 6, *Component Editing*, for information on how to override this setting for individual components.

Preferences - General Tab

In this tab, you set up selection rectangle behavior, mouse wheel behavior, autowiring behavior, and language.

Complete the following steps to set up parts preferences:

1. Choose **Options»Global Preferences** and click on the **General** tab that appears in the **Preferences** dialog box.
2. In the **Selection Rectangle** box, select either **Intersecting** or **Fully enclosed**. Refer to the *Setting Dragging Selection Options* section of Chapter 2, *Schematic Capture - Basics*, for more information.
3. In the **Mouse Wheel Behavior** box select one of:
   - **Zoom workspace**—Use mouse wheel to zoom in and out on workspace.
   - **Scroll workspace**—Use mouse wheel to scroll up and down the page.
4. Enable the **Show line to component when moving its text** and **Show line to original location when moving parts** checkboxes as desired.
5. In the **Wiring** box select the desired options to control the degree of automation used in wiring. Refer to the *Setting Wiring Preferences* section of Chapter 2, *Schematic Capture - Basics*, for more information.
6. Enable **Delete associated wires when deleting component** if you wish to delete wires that are connected to components being deleted. If this box is unchecked, wires connected to deleted components will remain.
7. In the **Language** box, select the desired locale. You must restart Multisim for the new setting to take effect.

Preferences - Simulation Tab

In this tab, you set up netlist error behavior, default graph and instrument background color, and positive phase shift direction.

Complete the following steps to set up parts preferences:

1. Choose **Options»Global Preferences** and click on the **Simulation** tab that appears in the **Preferences** dialog box.
2. In the **Netlist Errors** box, set the following as desired:
   - **When a netlist error occurs** drop-down list—Select one of **Prompt user**; **Cancel simulation/analysis**; **Proceed with simulation/analysis**.

- **When a netlist warning occurs** drop-down list—Select one of **Prompt user**; **Cancel simulation/analysis**; **Proceed with simulation/analysis**.

Refer to the *Netlist Errors and Warnings* section for more information.

3. In the **Graphs** box, set the following as desired:
   - **Default background color for graphs and instruments**—Select either **Black** or **White**. **Black** is suggested for onscreen viewing; **White** is suggested for printouts. Graphs that have already been run and instruments that have already been placed will not be affected by this setting.
4. Set **Positive Phase Shift Direction** as desired. This setting only affects the **Phase** parameter in AC sources.

Using the Sheet Properties Dialog Box

The **Sheet Properties** dialog box is used to set up the preferences for each sheet. These preferences are saved with the circuit file so that if the circuit is opened on another computer, it will use the same settings.

Note Refer to the *Using the Preferences Dialog Box* section for information on personal (user) preferences.

Complete the following steps to set sheet preferences:

1. Select **Options»Sheet Properties**.

   Or

   Right-click on a blank area of the workspace and choose **Properties** from the pop-up menu that appears. Do not select any elements (for example, components, wires) on the workspace, or the properties dialog box for the selected element will display instead.

   The **Sheet Properties** dialog box appears, offering you the following tabs:

   - **Circuit**—Where you set the color scheme and the display properties of workspace text
   - **Workspace**—Where you set the sheet size and properties
   - **Wiring**—Where you set the wire and bus options
   - **Font**—Where you select the font, font size and style for text elements on the circuit.
   - **PCB**—Where you set up options for your printed circuit board
   - **Visibility**—Where you enable and disable custom annotation layers.

2. Select the desired tab.
3. Set the desired customization options. The specific options and settings available in the tabs are described in the following sections of this chapter.
4. The **Save as Default** checkbox is normally enabled. If you do *not* wish to have the settings saved as the default, disable this checkbox. Default settings are those used when you create new circuits (**File»New**).
5. Click **OK** to save your changes. (If you wish to apply the changes without closing the dialog, click **Apply.**)

Sheet Properties - Circuit Tab

The options in this tab control the way your circuit and its components appear on the workspace, and the level of detail which appears. Multisim comes with several color schemes that affect the circuit window background color, wire color, and component color. You can also develop your own color scheme to meet your individual needs.

1. In the **Component** box, enable those items you want shown on the workspace. The results of enabling these options is shown in the preview area. Refer to the *Displaying Identifying Information about a Placed Component* section of Chapter 3, *Schematic Capture - Advanced Functions*, for information on how to override your choices for a particular component.

   For symbol and footprint pin names:

   - **Symbol Pin Names** checkbox—When fully checked, shows all symbol pin names; when not checked, shows no symbol pin names; when grey checked, shows symbol pin names for those components whose default behavior is to show them.
   - **Footprint Pin Names** checkbox—When fully checked, shows all footprint pin names; when not checked, shows no footprint pin names; when grey checked, shows footprint pin names for those components whose default behavior is to show them.

Note Pin names and numbers are made visible as detailed above; checking and unchecking in the **Visibility** tab of the **Design Toolbox** has no effect.

2. In the **Net Names** box, select how net names will be shown:
   - **Show All**—Check to display all net names on the workspace.
   - **Use Net-specific Setting**—Check to show net names as set in the **Net** dialog box. Refer to the *Modifying Net Names* section of Chapter 2, *Schematic Capture - Basics*, for more information.

- **Hide All**—Check to hide all net names on the workspace.

3. In the **Bus Entry** box, enable the **Show Labels** checkbox if you wish to display the busline name in circuits that contain buses.
4. In the **Color** box:
   - To use one of the built-in color schemes:
     - Choose the scheme from the drop-down list.
     - A representation of the scheme's settings appears in the preview box below the list.
   - To create a custom color scheme:
     - Choose **Custom** from the drop-down list.
     - Click on the colored button next to any item. A **Color** selector dialog box appears.
     - Click on the color you want to use for that item and click **OK**.
     - You are returned to the **Sheet Properties** dialog box.
     - The results of your choice appear in the preview area.
     - Repeat until all your color settings are made.

Sheet Properties - Workspace Tab

The options on this tab determine the appearance and behavior of the circuit window.

Multisim comes with standard sheet sizes that you can use to capture your circuit. You can modify any of the settings of these sizes to make your own custom sheet.

1. In the **Show** box, enable the **Show grid**, **Show page bounds** and **Show border** checkboxes as desired. The results appear in the preview area.
2. In the **Sheet size** box, select the desired size from the drop-down list.
3. Select either **Portrait** or **Landscape** in the **Orientation** box.
4. In the **Custom size** box, you can set the **Width** and **Height** as desired.

Sheet Properties - Wiring Tab

The options on this tab control the wire width, bus width and bus wiring mode options.

1. In the **Drawing Option** box, change the **Wire width** and **Bus width** as desired for current or subsequent wires and buses.
2. In the **Bus Wiring Mode** box, select either **Net** or **Busline**. Refer to the *Buses* section of Chapter 4, *Working with Larger Designs*, for more information.

Sheet Properties - Font Tab

Use this tab to set the font parameters for some or all of the elements in the workspace that contain text.

Caution Changing to larger fonts may cause labels to collide or exceed boundaries.

Complete the following steps to change the font for any text element of the circuit:

1. Select the **Font**, **Font Style** and **Size** for the desired elements.

   Multisim supports Unicode characters, which lets you use Cyrillic and Asian fonts. However, you must be sure to select a font that is Unicode-compatible, or the desired character will display as a rectangle.
2. Select the elements you want to change from the following options in the **Change All** section of the tab:
   - **Component RefDes**—The component's unique identifier (Reference Designator). For example, R22.
   - **Component Values and Labels**—The value being used by the component and identified in the **Value** tab of the component's properties dialog box. Refer to the *Viewing a Placed Component's Value/Model* section of Chapter 3, *Schematic Capture - Advanced Functions*, for information on viewing and modifying this value. The component label is assigned by Multisim and identified in the **Label** tab of the component's properties dialog box. Refer to the *Modifying Component Labels and Attributes* section of Chapter 2, *Schematic Capture - Basics*, for information on changing this label.
   - **Component Attributes**—Descriptive information you add to specific components using the component's properties dialog box.

Refer to the *Modifying Component Labels and Attributes* section of Chapter 2, *Schematic Capture - Basics*, for more information.

- **Footprint Pin Names**—The name assigned to a pin when it is transferred to PCB layout.
- **Symbol Pin Names**—The name assigned to a pin, for example, GND (for ground).
- **Net Names**—The name automatically assigned to a net when it is placed on your circuit. Refer to the *Modifying Net Names* section of Chapter 2, *Schematic Capture - Basics*, for information on changing this name.
- **Schematic Texts**—Notes that you add to your circuit using the **Place»Text** command. Refer to the *Adding Miscellaneous Text* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Comments and Probes**—Text found in comments that you add to the circuit from the **Place»Comment** command. Also for text found in probes that placed from the **Instruments** toolbar. Refer to the *Adding a Comment* section of Chapter 2, *Schematic Capture - Basics*, and the *Measurement Probe* section of Chapter 9, *Instruments*, for more information.
- **Busline Name**—The name that you assign to a busline. Refer to the *Buses* section of Chapter 4, *Working with Larger Designs*, for more information.

Note In the **Apply to** box, **Entire Circuit** changes the font for all instances of the elements selected in the **Change All** box. To change the font for a specific selection, right-click on the desired element and select **Font** from the pop-up that appears, or select it before you open the **Sheet Properties** dialog box, then enable the **Selection** radio button in the **Font** tab.

Sheet Properties - PCB Tab

This tab is used to set options used when exporting data for PCB layout.

1. In the **Ground Option** box, select **Connect digital ground to Analog ground** if you wish to make the digital and analog grounds the same when exporting to PCB layout packages.
2. In the **Export Settings** box, select the **Units** to use during export to PCB layout.
3. Enter the desired **Number of Copper Layers**. As this value increases, the number of copper layers (inner) increases, and is reflected in the list below the **Number of Copper Layers** field. This setting is used by Ultiboard to determine the default board setup.

Sheet Properties - Visibility Tab

This tab lets you add the custom annotation layers that are available in Multisim.

1. In the **Custom Layers** box, click **Add** to add custom annotation layers to the schematic.
2. Show/hide these layers from the **Visibility** tab in the **Design Toolbox**. Refer to the *Visibility Tab* section for more information.

Note Refer to the *Visibility Tab* section for information on annotation layers.

Design Toolbox

The **Design Toolbox** is used to manage various elements in the schematic.

Note Refer to the *Project Management and Version Control* section of Chapter 4, *Working with Larger Designs*,for information on the **Project View** tab.

Visibility Tab

The **Design Toolbox Visibility** tab lets you choose which layers to display on the current sheet on the workspace.

The **Schematic Capture** layers consist of:

- **RefDes**—This layer contains the reference designators for all elements on the workspace, for example, R1, U2A.
- **Label and Value**—This layer contains the label entered in the component's properties dialog box in the **Label** field in the **Label** tab. It also contains the component's value, for example, 1 kohm. Refer to the *Placed Component Properties* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.
- **Attribute and Variant**—This layer contains the attribute information entered in the component's properties dialog box in the **Attributes** fields in the **Label** tab. It also includes the **Variant Status**. Refer to the *Variants* section of Chapter 4, *Working with Larger Designs*, for more information.
- **Net Name**—This layer contains the netnames.
- **Pin Name**—This layer contains the pin names.
- **Pin Number**—This layer contains the pin numbers.
- **Bus Entry Label**—This layer contains the labels that appear at the point where a wire enters a bus, for example, "Ln 1".

The **Fixed Annotations** layers consist of:

- **ERC Error Mark**—This layer contains the marks that are placed on the schematic that indicate ERC error points. Refer to the *Electrical Rules Checking* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.
- **Static Probe**—This layer contains the static measurement probes that can be placed on the schematic. Refer to the *Measurement Probe* section of Chapter 9, *Instruments*, for more information.
- **Comment**—This layer contains any comments that you "pin" to the workspace. Refer to the *Adding a Comment* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Text/Graphics**—This layer contains any graphic elements that you place on the workspace.

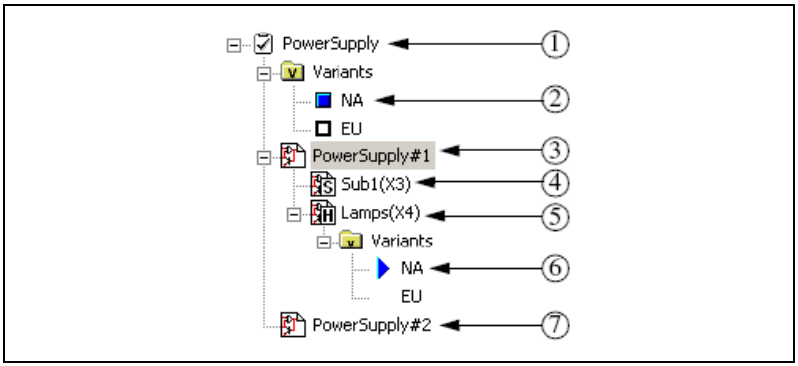
Custom Annotation layers can be added from the **Sheet Properties** dialog box. Refer to the *Sheet Properties - Visibility Tab* section for more information.

To hide a layer, disable the layer's checkbox.

To show a hidden layer, enable the layer's checkbox.

Hierarchy Tab

The **Design Toolbox**'s **Hierarchy** tab contains a tree that shows the files in the design that you have open.



1	Design Root	5	Hierarchical Blocks
2	Active Variant	6	Included in Active Variant
3	Page 1 of Multi-sheet	7	Page 2 of Multi-sheet
4	Subcircuit		

Refer to the *Variants*, *Flat Multi-Sheet Design*, and *Hierarchical Design* sections of Chapter 4, *Working with Larger Designs*, for more information.

Note Variants are not available in all versions of Multisim.

Pop-up Menus

Complete the following steps to view and use context-sensitive menus in the **Hierarchy** tab:

1. Right-click on an item. Depending on the item, different context-sensitive menus appear.
2. From the **Design Root** select:
   - **Close**—To close the entire design.
   - **Save**—To save the design.
3. From an active or in-active variant that branches from the **Design Root**, select:
   - **Set Variant Active**—To set the active variant. A blue box displays beside the active variant. Refer to the *Setting the Active Variant for Simulation* section of Chapter 4, *Working with Larger Designs*, for more information.
   - **Variant Manager**—To display the **Variant Manager** dialog box. Refer to the *Setting Up Variants* section of Chapter 4, *Working with Larger Designs*, for more information.
4. From a page of a multi-sheet design (in this example, PowerSupply#1), select:
   - **Open Window**—To open the page on the workspace.
   - **Close Window**—To close the page.
   - **Rename Page**—To change the multi-page's name.
5. From a subcircuit, select:
   - **Open Window**—To open the subcircuit on the workspace.
   - **Close Window**—To close the subcircuit.
   - **Map Variants**—To display the subcircuit's properties dialog box, where you can assign its variant status. Refer to the *Assigning Variant Status to Components* section of Chapter 4, *Working with Larger Designs*, for more information.
   - **Rename Subcircuit**—To change the subcircuit's name.
6. From a hierarchical block, select:
   - **Open Window**—To open the hierarchical block on the workspace.
   - **Close Window**—To close the hierarchical block.

- **Map Variants**—To display the hierarchical block's properties dialog box, where you can assign its variant status. Refer to the *Assigning Variant Status to Components* section of Chapter 4, *Working with Larger Designs*, for more information.

7. From a variant that branches from a hierarchical block, select:
   - **Variant Manager**—To display the **Variant Manager** dialog box. Refer to the *Setting Up Variants* section of Chapter 4, *Working with Larger Designs*, for more information.
   - **Exclude from Active Variant**—To exclude this variant from the active variant. If desired, you can exclude all variants in a hierarchical block from the active variant.
   - **Include in Active Variant**—To include this variant in the active variant. A blue triangle appears beside the included variant. You can only include one variant at a time in the active variant.

Customizing the Interface

The Multisim user interface is highly customizable. Separate customizations can be applied whenever a different type of sheet becomes active. For example, the toolbars and docking windows can be re-configured as you move from a circuit sheet to a description sheet.

Toolbars can be docked in various positions and orientations. The contents of the toolbars can be customized. New toolbars can be created. The menu system is fully customizable, including all pop-up menus for the various object types.

As well, the keyboard shortcut system is customizable. This allows for any keys or key combinations to be assigned to any command that can be placed in a menu or on a toolbar.

Complete the following steps to customize the interface:

1. Select **Options»Customize User Interface**.
2. Make changes as detailed in the following sections:
   - *Commands tab*
   - *Toolbars tab*
   - *Keyboard tab*
   - *Menu tab*
   - *Options tab*
   - *Customization Pop-up Menus*.

Commands tab

The **Commands** tab in the **Customize** dialog box is used to add commands to menus and toolbars.

Complete the following steps to add a command to a menu or toolbar:

1. Drag it from the **Commands** list to the desired menu or toolbar. When a command is selected in the **Command** list, its description is displayed in the **Description** field.
2. If you do not see the command that you require, click on another selection in the **Categories** list to display more commands.
3. Click **Close** when customizations are complete.

To remove a command from a menu or toolbar, right-click on it and select **Delete** from the pop-up that appears. The **Customize** dialog box must be open when you do this.

To change the position of a command that is in a menu or toolbar, drag it to its new location. The **Customize** dialog box must be open when you do this.

Toolbars tab

The **Toolbars** tab in the **Customize** dialog box is used to show or hide toolbars, and to add new custom toolbars.

Complete the following steps to use the features in this tab:

1. To display a toolbar, switch on the checkbox beside the desired toolbar in the **Toolbars** list.
2. Switch off a checkbox to hide a toolbar.

Note You cannot switch off the **Menu** bar.

3. Use the following buttons and checkboxes as desired:
   - **Reset All**—Displays the **Reset Toolbars** dialog box, where you select whether to reset the currently selected toolbars, or all toolbars. You are prompted to select the configuration file you wish to use, for example, “default.ewcfg”.
   - **New**—Displays the **Toolbar Name** dialog box, where you enter the name for a new toolbar. When you click **OK**, a new toolbar with the name that you entered is created. Follow the steps in the *Commands tab* section to add buttons to the toolbar.

- **Rename**—Use to rename a toolbar that you have created yourself. You cannot rename toolbars that are included in Multisim by default, for example, **Components**, **Menu Bar**.
- **Delete**—Use to delete the selected toolbar. You cannot delete toolbars that are included in Multisim by default, for example, **Main, View, Components**.
- **Show text labels**—Select this checkbox to show the text labels (for example, "Save") in the toolbar, along with the command's icon.
- **Lock all docked toolbars**—Select this checkbox to lock all docked toolbars in place.

4. Click **Close** when customizations are complete.

Keyboard tab

The **Keyboard** tab is used to set up keyboard shortcuts.

Complete the following steps to set up keyboard shortcuts:

1. Choose a category from the **Category** drop-down list and the desired command from the **Commands** drop-down list. If a shortcut is already assigned, it appears in the **Current Keys** field.
2. Enter a new shortcut in the **Press New Shortcut Key** field.
3. Click **Close** when customizations are complete.

Menu tab

The **Menu** tab is used to modify the various context-sensitive menus that appear when you right-click from various locations in Multisim.

Complete the following steps to display the desired menu:

1. Select the desired menu set from the **Select Context Menu** drop-down list.
2. Right-click on the menu that appears and edit as desired.
3. Select the desired menu effects using the **Menu animations** drop-down list and the **Menu shadows** checkbox.

Options tab

The **Options** tab in the **Customize** dialog box is used to set up toolbar and menu options.

To set up menu and toolbar options, switch the checkboxes on or off as desired.

Customization Pop-up Menus

To customize the appearance of toolbar buttons and menu items, a pop-up menu is available when the **Customize** dialog box is open.

Complete the following steps to display the above pop-up:

1. Be sure you have the **Customize** dialog box open.
2. Right-click on either a menu item or toolbar and select the desired option from the pop-up menu that appears.
3. When you select **Button Appearance**, the **Button Appearance** dialog box appears, where you can change the appearance of the selected toolbutton.

2

Schematic Capture - Basics

This chapter describes the basic functions involved in creating a circuit in the circuit window. It explains the fundamental steps in circuit creation, but is not intended to describe all the potential aspects of circuit design. For example, you should look to other chapters for details on the component database and instructions on editing components.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Introduction to Schematic Capture

Schematic capture is the first stage in developing your circuit. In this stage you choose the components you want to use, place them on the circuit window in the desired position and orientation, wire them together, and otherwise prepare your design. Multisim lets you modify component properties, orient your circuit on a grid, add text and a title block, add subcircuits and buses, and control the color of the circuit window background, components and wires.

Note Refer to the *Setting Schematic Capture Preferences* section of Chapter 1, *User Interface*, for information on how to set up your workspace.

Working with Multiple Circuit Windows

You can open as many circuits as you want at the same time. Each circuit appears in its own circuit window. The active circuit window is, as in other Windows applications, the window with the highlighted title bar. You can use the Window menu to move from circuit window to circuit window or just click on the tab at the bottom of the workspace for the circuit you want to see.

Each window is distinct, and can have its own preferences settings, set of components and so on. You can copy, but not move, a component or instrument from one circuit window to another.

Selecting Components from the Database

The first step in schematic capture is placing the appropriate components on your circuit window.

You can use the following methods to locate a component in the database:

- use the **Components** toolbar to browse through all the component groups. Refer to the *Using the Place Component Browser* section for more information.
- search for a specific component group/family in the database. Refer to the *Searching for Components* section of Chapter 5, *Components*, for more information.

Note There are multiple "levels" of the component database. Refer to the *Database Levels* section of Chapter 5, *Components*, for more information.

The first of these choices is normally used. Each button on the **Components** toolbar corresponds to a group of components with similar functionality. Click one of these buttons to open the place component browser (that is, the **Select a Component** dialog box) with the contents of that button's group displayed.

Multisim provides the unique concept of virtual components. Virtual components have a symbol and a model, but no footprint and hence are not "real" in the sense that they cannot be purchased. They are included for simulation flexibility. Virtual component families are color coded green in the **Select a Component** dialog box.

Placing Components

The component browser is used to select parts from the component databases and place them on a circuit. Parts are organized by database, group, and family (for example, **Master Database**, **Sources Group**, **Power Sources Family**). Filters are provided as appropriate to narrow lists based on value range and tolerance where applicable. Type-ahead allows you to type a few characters to jump to the component you are looking for. Search capabilities allow you to find parts using generalized wildcard searches throughout all the databases.

Using the Place Component Browser

The procedure described in this section applies for the placement of most components. Refer to the *Placing Resistors, Inductors or Capacitors* section for information on the placement of R, L, or C components. Refer to the *Multi-section Components* section for information on the placement of multi-section components.

Complete the following steps to place a component:

1. Click on the desired group in the **Components** toolbar, for example, **Transistors**. The **Select a Component** dialog box appears with the selected component group displayed.

   Alternatively, you can display the **Select a Component** dialog box by choosing **Place»Component** and selecting the desired group from the **Group** drop-down list.

   Or

   Right-click in the workspace and select **Place Component** from the pop-up that appears.

Note The **Select a Component** browser is also referred to as the place component browser.

2. The default database that displays in the browser is the **Master Database**. If you wish to select a component from either the **Corporate Database** or **User Database**, you must select that database from the **Database** drop-down list before selecting a component. Once changed, the database will remain as selected for subsequent part placements.
3. Click on the desired component family in the **Family** list.
4. Click on the desired component in the **Component** list.

Tip To make your scroll through the **Component** list faster, type the first few characters of the component's name in the **Component** field. As you type, matches are displayed in the top of the **Component** list. If you make a mistake, you can use the Backspace key to remove one character at a time.

Note The **Function** area shows any available information about the component selected in the **Component** list.

5. Select the desired model and model manufacturer in the **Model manuf./ID** area.

6. Selected the desired footprint in the **Footprint manuf./Type** list.

   Some virtual components (for example, power sources) do not have footprints available. Refer to the *Real Components* section of Chapter 3, *Schematic Capture - Advanced Functions*, for a discussion of "real" vs. virtual components.

   The **Hyperlink** field is intended to contain, for example, the internet address for the component's manufacturer. The contents of this field can be edited in the **Components** tab of the **Database Manager**. Refer to the *Modifying User Field Titles and Content* section of Chapter 5, *Components*, for more information. If you want to go to the link, hover your cursor it, hold down the <Ctrl> key and click your mouse.

7. Click **OK** to confirm that this is the component you want to place. The browser closes and the cursor on the circuit window changes to a ghost image of the component. This indicates that the component is ready to be placed.

Note If you are placing a component that is one part of a multi-section component such as a quad 2-input NAND gate, a dialog box displays, where you specify which of the sections you want to place. Refer to the *Multi-section Components* section for more information about the dialog's functionality.

8. Move your cursor to the location where you want to place the component. The workspace automatically scrolls if you move your cursor to its edges.

9. Click on the circuit window where you want the component placed. The component's symbol and labels appear (unless you have specified that they are not to be displayed, as explained in the *Displaying Identifying Information about a Placed Component* section of Chapter 3, *Schematic Capture - Advanced Functions*), as well as a unique RefDes made up of a letter and number. The letter represents the type of component and the number is a sequential number that indicates the order in which the components were originally placed. For example, the first digital component has the RefDes "U1", the next is "U2", the first inductor has the RefDes "L1", and so on.

   If you have **Return to Component Browser after Placement** selected in the **Parts** tab of the **Preferences** dialog box, the **Select a Component** browser reappears. Refer to the *Preferences - Parts Tab* section of Chapter 1, *User Interface*,for more information.

If the component you place is a virtual component, it appears in a different color from real components. This color is set in the **Sheet Properties**

dialog box. Refer to the *Sheet Properties - Circuit Tab* section of Chapter 1, *User Interface*, for more information.

Placing Resistors, Inductors or Capacitors

The procedure used to place resistors, inductors or capacitors is quite similar to the procedure used to place other types of components. However, there are some differences.

When placing any of these components, you can choose any combination of: the component's value (for example, resistance); type (for example, carbon film); tolerance; footprint and manufacturer.

Caution If you are placing a component that will be exported to PCB layout, and become part of a **Bill of Materials**, you must be careful that the combination of values that you select in the **Select a Component** dialog box are available in a real-world, purchaseable component.

Complete the following steps to place an R, L or C component:

1. Click on the **Basic** group in the **Components** toolbar. The **Select a Component** dialog box appears with the **Basic** component group displayed.

   Or

   Choose **Place»Component** to display the **Select a Component** dialog box, and select the **Basic** group from the **Group** drop-down list.

   Or

   Right-click in the workspace and select **Place Component** from the pop-up that appears. Select the **Basic** group from the **Group** drop-down list.
2. Select the **Master Database** in the **Database** drop-down list.
3. Click on the desired component family in the **Family** list, for example **Resistor**.
4. Type the value of the component that you want to place in the field at the top of the **Component** list. You can type any value—it does not need to appear in the list to be placed on the schematic. Refer to the *Acceptable Unit Letters* section for information about what letters can be used in this field.
5. Optionally, enable the **Save unique component on placement** checkbox. When this is enabled, any component with a unique combination of the values in the various fields of the **Select a Component** dialog box is saved.

Note R, L, and C components found in the **Master** database are in fact a combination of the value, component type, tolerance, and footprint manufacturer/type. Whenever a unique combination of these values is selected in the **Select a Component** dialog box, and the **Save unique component on placement** checkbox is enabled, the combination of these values is saved in the **Corporate** database. Consequently, if the **Corporate** dB is deleted, these components no longer appear in the **Master** database. Refer to the *Backing up the Corporate Database* section of Appendix B, *Archiving Data* for important information.

6. Select the desired **Component Type**. If you do not see the desired type in the list, you can type it manually.
7. Select the desired **Tolerance**. If you do not see the desired tolerance in the list, you can type it manually. This list does not appear for potentiometers, variable inductors, or variable capacitors.
8. Selected the desired footprint in the **Footprint manuf./Type** list. If you are placing a part for simulation only, select <no footprint>. If you intend to export the schematic to PCB layout, select a value in this list.

   The **Hyperlink** field is intended to contain, for example, the internet address for the component's manufacturer. The contents of this field can be edited in the **Components** tab of the **Database Manager.** Refer to the *Modifying User Field Titles and Content* section of Chapter 5, *Components*, for more information. If you want to go to the link, hover your cursor over it, hold down your <Ctrl> key and click your mouse.
9. To confirm that this is the component you want to place, click **OK**. The browser closes and the cursor on the circuit window changes to a ghost image of the component you wish to place. This indicates that the component is ready to be placed.
10. Move your cursor to the location where you want the component placed. The workspace automatically scrolls if you move your cursor to its edges.
11. Click on the circuit window where you want the component placed.

    If you have **Return to Component Browser after Placement** selected in the **Parts** tab of the **Preferences** dialog box, the **Select a Component** browser reappears. Refer to the *Preferences - Parts Tab* section of Chapter 1, *User Interface*, for more information.

Note You can also create resistors, inductors or capacitors using Multisim's **Component Wizard**, however, R, L, and C components created in this way will only contain basic simulation model information. Those that are placed from the **Master Database** as described in this procedure have additional SPICE simulation parameters that can be viewed from the **Value** tab of the placed component's properties dialog box. When you are

placing components that were created using the **Component Wizard**, you must select them from either the **Corporate** or **User Database**.

To edit a placed R, L, or C component, double-click on the component and click the **Value** tab.

Acceptable Unit Letters

Some manufacturers do not use standard units to describe their R, L, or C components in price lists and specifications. To accommodate this, Multisim lets you enter any number in scientific notation (with or without an exponent), followed by an optional unit letter for the value of resistors, inductors, and capacitors.

The following table describes the acceptable unit letters.

Unit Letter	Description
f or F	femto (1e-15)
p	pico (1e-12)
n or N	nano (1e-9)
u or U or µ	micro (1e-6)
m	milli (1e-3)
k or K	kilo (1e+3)
M or MEG or meg	mega (1e+6)
g or G	giga (1e+9)
t or T	tera (1e+12)
P	peta (1e+15)

You can also use the unit letter as a subsitute for the decimal point in a number, so 1M3 = 1.3M, for example. Do not use an exponent in this case.

Letters with multiple acceptable capitalizations are auto-corrected to match the standard for that unit. For example, if you type 1.3g, it is converted to 1.3G.

Exception: “k” is normally lower case. However, if it is used in place of a decimal point as in the example “1k3”, it is converted to upper case, as in “1K3”.

Any letters after the suffix are ignored. For example, “1.4kohms” = “1.4k” and “1.3me” = “1.3m”.

Acceptable values include:

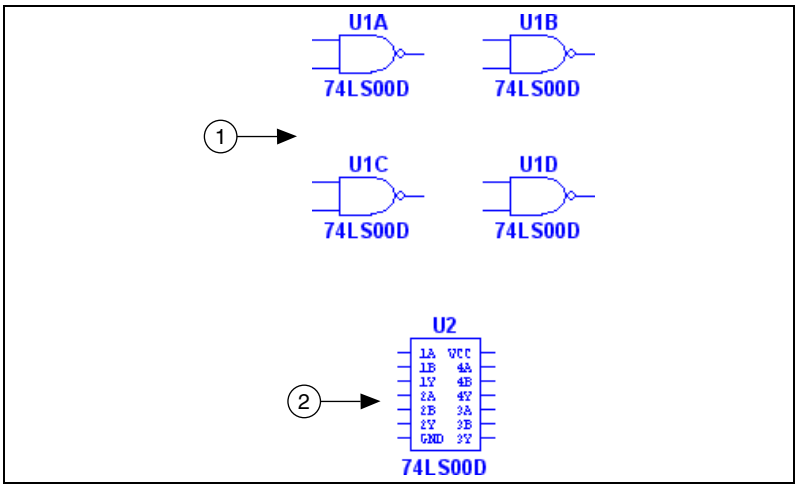
- 1.3
- 1.3k
- 1k3
- 1.3kohms
- 1.3e4
- 1.3e-4
- 1.3e-4meg

Multi-section Components

Certain non-virtual part symbols do not have a one-to-one correspondence with a footprint. Rather, several component symbols may correspond to a single physical package. An example is a quad 2-input NAND gate such as the Texas Instruments 74LS00D. For this particular device, up to four component symbols on a schematic diagram may correspond to a single part for the purposes of PCB layout.

To place a multi-section part, select the individual part (in this case, a NAND gate) from the component browser. Immediately prior to placing the component, a section chooser pops up that lists parts that have free sections, or allows you to start placing a new part. You must select one of these sections for placement. Once placed, optimizing facilities are provided to enable multi-section parts to be efficiently packed into chips. A **Spare Gates Report** is available to show sections not used on multi-section parts in the circuit. Refer to the *Spare Gates Report* section of Chapter 12, *Reports*, for more information.

As well as being available in components with individual sections as described above, some multi-section TTL and CMOS components are available in a single-footprint format containing all of the devices. These components’ parts families are found in the place component browser, and are appended with “_IC”, as in the part family “74LS_IC”.



1	Individual 74LS00D Gates	2	Single-footprint 74LS00D IC

In the above figure, (1) shows 74LS00D gates placed as individual components. These four gates were place from the 74Ls component family, and will be contained in one component package when the circuit is exported for PCB layout. It also shows a single-footprint 74LS00D IC containing four NAND gates (2). This was placed from the 74LS_IC component.

Complete the following steps to place a multi-section part with a single footprint:

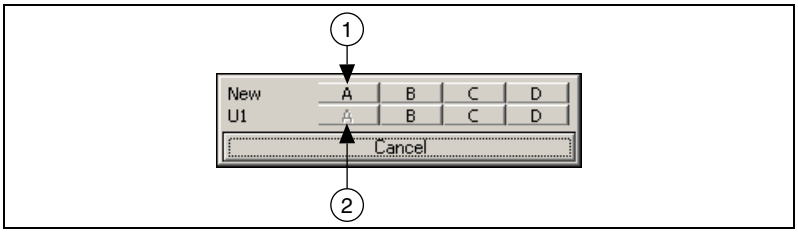
1. Select **Place»Component** and navigate to the desired **Group**, **Family** and **Component**.
2. Click to place the IC on the workspace.

Note Single-IC components are not available for all multi-section components.

Complete the following steps to place one device of a multi-section part:

1. Select **Place»Component** and navigate to the desired **Group**, **Family** and **Component**.
2. Click **OK**. If this is the first multi-section component of this type (for example, 74LS00D) being placed on the circuit, a dialog box appears as shown in the figure below:

If there are already other multi-section components placed, the dialog box will be similar to the following:



1 Darker Text 2 Greyed-out Text

Darker text (1) indicates available sections for placement. Greyed-out text (2) indicates this section has already been placed.

3. Click on an available section to place it.

Note It is a good idea to place all the sections of a multi-section device (U1 in this example) before going to a new multi-section device of the same type. Only instances of the same component type are shown as available to place. In the above example, U1 and "New" are both quad 2-input 74LS00D ICs. If there are other quad 2-input NAND devices in the circuit, for example 74LS01N, they will not be displayed.

4. Click on any available section. The dialog box closes and a ghost image of the device is attached to your cursor.
5. Click to place the device in the desired location.
6. If you selected either **Continuous placement for multi-section part only** or **Continuous placement** in the **Parts** tab of the **Preferences** dialog box, the dialog box re-appears.
7. Continue placing parts from this dialog. When finished, press <Esc> to exit.

Rotating/flipping a part during placement

Complete the following steps to rotate or flip a part during placement:

1. Select a part as detailed in the *Using the Place Component Browser* section.
2. As you are dragging the "ghost" image of the part that you are placing, press one of the following key combinations:
   - **Ctrl-R**—rotates the component 90 degrees clockwise.
   - **Ctrl-Shift-R**—Rotates the component 90 degrees counter-clockwise.

- **Alt-X**—Flips the component horizontally.
- **Alt-Y**—Flips the component vertically.

Other Buttons

To display the **Search Component** dialog box, click the **Search** button. Refer to the *Searching for Components* section of Chapter 5, *Components*, for more information.

To display the **Component Detail Report** dialog box, click the **Detail Report** button. Refer to the *Component Detail Report* section of Chapter 12, *Reports*, for more information.

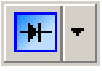
To display the **Model Data Report** dialog box, click the **Model** button. Refer to the *Model Data Report* section of Chapter 12, *Reports*, for more information.

Placing Virtual Components

Use the **Virtual** toolbar to place virtual components on your workspace.

Complete the following steps to place a virtual component:

1. Click on the desired button in the **Virtual** toolbar.

Button	Description
	Show Power Source Components button. Displays the **Power Source Components** toolbar, which contains buttons that let you place different virtual Power Source components.
	Show Signal Source Components button. Displays the **Signal Source Components** toolbar, which contains buttons that let you place different virtual Signal Source components.
	Show Basic Components button. Displays the **Basic Components** toolbar, which contains buttons that let you place different virtual Basic components.
	**Show Diode Components** button. Displays the **Diodes Components** toolbar, which contains buttons that let you place different virtual diodes.

Button	Description
	Show Transistor Components button. Displays the **Transistor Components** toolbar, which contains buttons that let you place different virtual transistors.
	Show Analog Components button. Displays the **Analog Components** toolbar, which contains buttons that let you place different virtual Analog components.
	Show Miscellaneous Components button. Displays the **Miscellaneous Components** toolbar, which contains buttons that let you place miscellaneous virtual components.
	Show Measurement Components button. Displays the **Measurement Components** toolbar, which contains buttons that let you place different virtual Measurement components.
	Show Rated Virtual Components button. Displays the **Rated Virtual Components** toolbar, which contains buttons that let you place different virtual components that are found in the Rated Virtual component group. This component group contains a number of virtual components that can be rated to “blow” if pre-set tolerance(s) are exceeded when the circuit is simulated. These tolerances are set in the **Values** tab of each component’s properties window. This feature is not available in all versions of Multisim.
	Show 3D Components button. Displays the **3D Components** toolbar, which contains buttons that let you place different virtual 3D components. These components function normally when the circuit is simulated, but appear like the real component on the circuit schematic. This feature is not available in all versions of Multisim.

Note Refer to the *Virtual component toolbars* section for information about the various toolbars that appear when buttons on the **Virtual** toolbar are pressed.

2. From the toolbar that displays, click on the desired virtual component. The cursor changes to a “ghost” image of the component you wish to place.

Tip You can click on the down-arrow beside any of the buttons in the **Virtual** toolbar to display the components available for that button and select the desired component directly from the drop-down list that appears.

3. Click the workspace in the desired location to place the virtual component.

Virtual component toolbars

This section describes the virtual components that can be placed from the various virtual component toolbars that are accessed by pressing the corresponding button on the **Virtual** toolbar. Refer to the *Placing Virtual Components* section for more information.

Power Source Components

The buttons (from left to right) in the **Power Source Components** toolbar place the following virtual components: AC Power Source; DC Power Source; Digital Ground; Ground; 3 Phase Voltage Source Delta; 3 Phase Voltage Source Wye; VCC Supply; VDD Supply; VEE Supply; VSS Supply.

Signal Source Components

The buttons (from left to right) in the **Signal Source Components** toolbar place the following virtual components: AC Current Source; AC Voltage Source; AM Source; Clock Current Source; Clock Voltage Source; DC Current Source; Exponential Current Source; Exponential Voltage Source; FM Current Source; FM Voltage Source; PWL Linear Current; PWL Linear Voltage; Pulse Current Source; Pulse Voltage Source; Thermal Noise Source.

Basic Components

The buttons (from left to right) in the **Basic** toolbar place the following virtual components: capacitor; coreless coil; inductor; magnetic core coil; non-linear transformer; potentiometer; normally open relay; normally closed relay; combination relay; resistor; audio transformer; miscellaneous transformer; power transformer; transformer; variable capacitor; variable inductor; pullup resistor; voltage controlled resistor.

Diodes

The buttons (from left to right) in the **Diodes** toolbar place the following virtual components: diode; zener diode.

Transistor Components

The buttons (from left to right) in the **Transistor Components** toolbar place the following virtual components: BJT NPN 4T; BJT NPN; BJT PNP 4T; BJT PNP; GaASFET N; GaASFET P; JFET N; JFET P; several enhancement and depletion mode NMOSFETs and PMOSFETs.

Analog Components

The buttons (from left to right) in the **Analog Components** toolbar place the following virtual components: Comparator; 3 Terminal Op-amp; 5 Terminal Op-amp.

Miscellaneous Components

The buttons (from left to right) in the **Miscellaneous Components** toolbar place the following virtual components: 555 Timer; Analog Switch; Crystal; DCD Hex; Current Rated Fuse; Lamp; Monostable; Motor; Optocoupler; Phase Locked Loop; 7 Segment Display Common Anode; 7 Segment Display Common Cathode.

Measurement Components

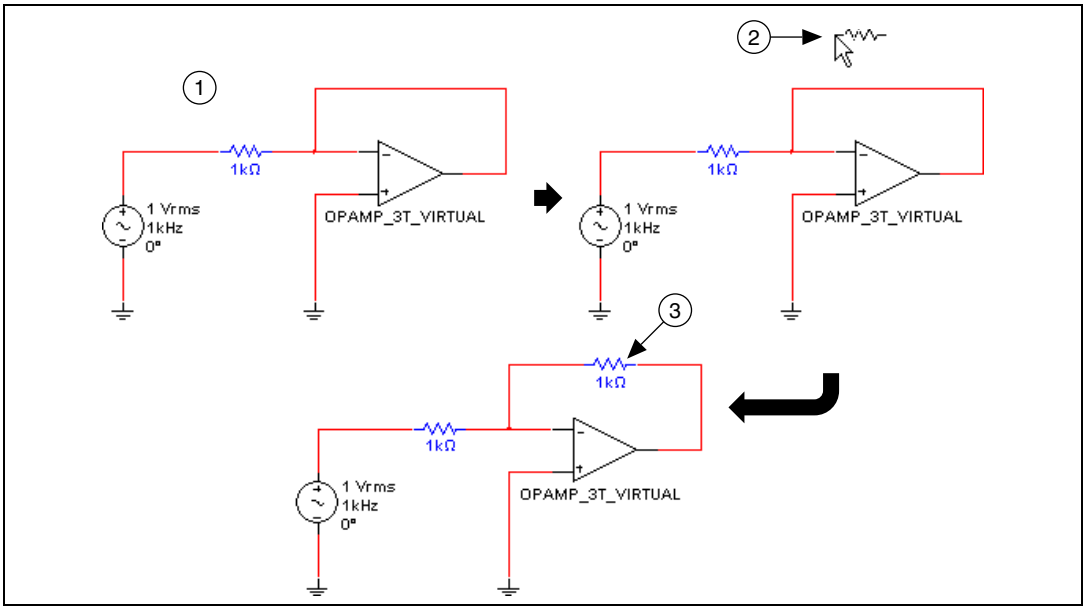
The buttons (from left to right) in the **Measurement Components** toolbar place the following virtual components: Ammeter (4 configurations); Digital Probe (5 colors); Voltmeter (4 configurations).

Using the In Use List

Each time you place a component, it is added to the **In Use List**. To place another instance of any currently placed component, select it from the list and click to place it in the desired location.

Drop-Placing Two-Pinned Components

Once you have wired a circuit, you can drop two-pinned components such as resistors directly onto a wire.



1 Original Circuit	2 New Component	3 Placed Component

Complete the following steps to drop a two-pinned component onto a wire in the original circuit in the above figure (1):

1. Select the desired two-pinned component as described in the *Using the Place Component Browser* section.
2. Move the "ghost" image of the component to the desired wire as shown in (2) in the above figure.
3. Release the mouse to "break" the wire and place the component in the break, as shown in (3) in the above figure.

Selecting Placed Components

You can select a component that has been placed on the workspace by either clicking on it or dragging the mouse to make a selection.

To select a component using the mouse, left-click on its center.

Tip If you wish to select a component's label or other text, but not the entire component, do not click on the center of the component—click directly on the text.

Complete the following steps to select a component by dragging the mouse:

1. Click and hold the left mouse button and drag the cursor to form a selection rectangle on the workspace that contains the component to be selected.

Note You can set whether you must enclose all or part of a component in the selection rectangle. Refer to the *Setting Dragging Selection Options* section for more information.

2. Release the mouse. The component is selected.

Tip You can select individual elements in a component like the RefDes, label and so on. To move the selection to another element of the component, use the <Tab>key.

Setting Dragging Selection Options

You can use the **Selection Rectangle** box in the **General** tab of the **Preferences** dialog box to set how part selection by mouse dragging functions.

If you select:

- **Intersecting**—The part is selected when a portion of it is contained in the dragged rectangle.
- **Fully enclosed**—The entire part, including all text and labels must be in the dragged rectangle for part to be selected.

Tip To toggle between the **Intersecting** and **Fully enclosed** modes, press and hold the Z key before dragging the mouse.

Moving a Placed Component

To move a placed component to another location, do one of the following:

- drag the component.
- select the component and press the arrow keys on your keyboard to move it up, down, or to either side in increments of one grid space.

Note Moving a component automatically adjusts its wiring to an appropriate configuration, unless you have disabled the **Autowire on move** option. Refer to the *Setting Wiring Preferences* section for more information.

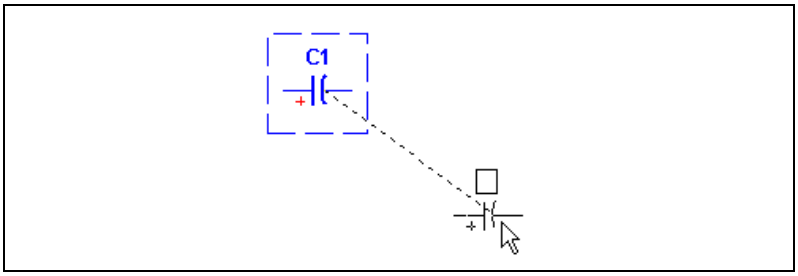
Complete the following steps to move a component by dragging:

1. Click and hold the left mouse button on the desired component. The component is selected with a dashed line.

Note A component's symbol and labels (for example, RefDes, Value) can be moved independently or together—if you plan to move the component, be sure the whole component is selected, not just its label. Refer to the *Selecting Placed Components* section for more information.

2. Drag the component to the desired location.

   As shown in the figure below, as you drag the mouse, a ghost image of the selected part appears attached to the cursor, along with a dashed line that connects the ghost image to the selected component's original location.



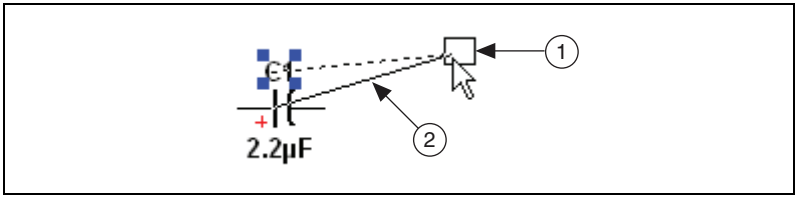
3. Release the mouse button when the ghost image is in the desired location.

Complete the following steps to move a component's label:

1. Click and hold the left mouse button on the desired label.
2. Drag the label to the desired location.

   As you drag the mouse, a ghost image of the selected label appears attached to the cursor, along with a dashed line that connects the ghost image to the selected label's original location, as shown in (1) in the figure below.

A solid line connects the ghost image to the component to which it belongs, as shown in (2) in the figure below. This is very useful in circuits that have a large number of components.



1 Ghost Image	2 Solid Line

3. Release the mouse button when the ghost image is in the desired location.

Copying a Placed Component

Complete the following steps to copy a placed component:

1. Select the desired component and choose **Edit»Copy**.

   Or

   Right-click on the desired component, and, from the pop-up menu that appears, choose **Copy**.

2. Select **Edit»Paste**.

   Or

   Right-click on the workspace select **Paste** from the pop-up menu that appears.

3. The cursor shows a “ghosted” version of the copied component. Click at the location where you want the copied component placed.

Note You can also copy a component using the Windows control keys for cut `<Ctrl-X>`, copy `<Ctrl-C>` and paste `<Ctrl-V>`.

Replacing a Placed Component

You can easily replace a placed component by using the **Replace** button in the component’s properties dialog box. The most common use for this feature is when you wish to start a circuit with virtual components, and then replace them with the nearest real components.

Complete the following steps to replace a placed component:

1. Double-click on the component you want to replace. The component's properties dialog box appears.
2. Click **Replace**. The **Select a Component** browser appears.
3. Select a new component and click **OK**. The new component appears on the circuit window in the place of the previous one. Refer to the *Using the Place Component Browser* section for more information about the component browser.

Note If you wish to replace multiple selected parts with the same quantity of one specific part, select **Tools»Replace Components**. Refer to the *Tools Menu* section of Appendix A, *Menus and Commands*, for more information.

Controlling Component's Color

The default color is controlled in the **Sheet Properties** dialog box. Refer to the *Sheet Properties - Circuit Tab* section of Chapter 1, *User Interface*, for more information.

Complete the following steps to change the color of a placed component:

1. Right-click on the component and choose **Change Color** from the pop-up menu that appears. You are presented with a color palette.
2. Choose a color and click **OK** to apply it to the selected item.

Wiring Components

A basic wire can be created by clicking on any one of a part's symbol pins. This creates a wire that can then be routed to either another symbol pin, or to another wire. If routed to a wire, when placed a *junction* is automatically created to differentiate between two wires crossing and two wires connected.

Wires can also be started by double-clicking anywhere on the circuit. This creates a junction at that location and starts wiring from that point. If a symbol's pins are dropped onto either a wire or another pin, a connection is automatically made.

Each time a wire is placed it either creates a new *net*, or joins an existing one. A *net* is a collection of wires all of which define a common electrical potential. The term net is typically used to describe this concept when discussing PCB layout. This concept is equivalent to the concept of a *node* when discussing SPICE-based circuit simulation.

Nets are typically assigned the next available small integer value. If two nets are merged by wiring them together, there are rules used to decide which of the two names the new net will bear, however in general the smaller-numbered net wins. You may also manually assign a name to a net.

Within a single-page of a circuit, a net may be manually renamed to be the same name as another on the same page. In this case, the two nets are merged together. This is called *virtual wiring* and may be used to reduce the complexity of circuits. Except for special reserved nets, virtual wiring may not be used across pages or across levels in the hierarchy. Refer to the *Virtual Wiring* section for more information.

Certain pre-defined named nets are considered global across an entire design. That is to say, anytime a net at any level in the hierarchy or on any page is re-named to one of these reserved nets, it joins this net. These reserved nets are 0, GND, VCC, VDD, VEE, and VSS. Net 0 corresponds to analog ground, and is the reference for all voltages during simulation. GND is a digital ground (as it is common for the purposes of PCB layout to wish to isolate these two ground nets).

These reserved nets are most often used in conjunction with *hidden symbol pins*. These are pins that are not shown on a schematic, as they and their accompanying wires would clutter the schematic to too great an extent, but are nonetheless connected for the purposes of simulation and layout. For example, a TTL digital AND gate would be connected to GND and VCC via hidden pins.

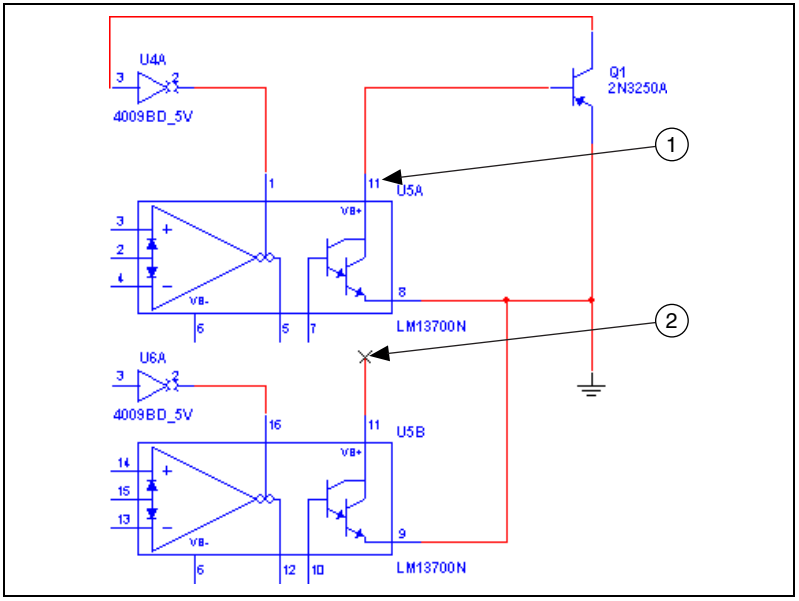
Wiring Components Automatically

Complete the following steps to wire two components together automatically:

1. Click on a pin from the first component to start the connection. Your pointer turns into a crosshair (✦).
2. Move the mouse. A wire appears, attached to your cursor.
3. Click on a pin on the second component to finish the connection. Multisim automatically places the wire, which snaps to an appropriate configuration (unless you have disabled the “autowire on connection” option, as described in the *Setting Wiring Preferences* section). The wire is numbered as a net. After a wire is connected between two pins the cursor returns to its normal mode and is ready for the next command.

Tip If the connection was not successful, you may be trying to place the wire too close to other surrounding components. Try to make the connection at a slightly different location, or use manual wiring. Refer to the *Wiring Components Manually* section for more information.

When wiring a component with multiple sections, an "X" indicates that a common pin has been connected in another section of the component. In the figure below, U5A and U5B are sections of a Dual, Current Controlled Operational Transconductance Amplifier. These two sections share a common pin on the IC (pin 11, VB+). (1) shows Pin 11 connected to Q1 from component U5A (section A of U5). In the same figure (2), Pin 11 of U5B (section B of U5) is marked with an "X". This indicates that pin 11 is connected in another section of U5.



1	Connected Pin	2	Common Pin Connection

To delete a wire, click on it and press Delete on your keyboard or right-click on it and choose **Delete** from the pop-up menu that appears.

Autowire of Touching Pins

You have the option to make an automatic connection if a component pin is touching a wire, another pin or a junction.

To enable/disable this option, select or deselect the **Autowire when pins are touching** checkbox in the **General** tab of the **Preferences** dialog box.

Complete the steps below to automatically connect a component to a wire:

1. Drag the component so that the desired pin is in contact with the desired wire.
2. Release the mouse button. A junction is automatically placed at the contact point between the component's pin and the wire.
3. Optionally, drag the component to a new location. The component stays connected to the wire.

Complete the steps in the example below to connect two components:

1. Drag one component into pin-to-pin contact with another component.
2. Release the mouse button. A junction is automatically placed at the contact point.
3. Optionally, drag a component to a new location. The two components stay connected.

Wiring Components Manually

If you want to select the precise path a wire will take on a schematic, use the procedure documented in this section.

Complete the following steps to wire two components together manually:

1. Click on a pin from the first component to start the connection. Your pointer turns into a crosshair (✦).
2. Move the mouse. A wire appears, attached to your cursor.
3. Control the flow of the wire by clicking on points as you move the mouse. Each click "fixes" the wire to that point.

Note By default, Multisim "skips over" (avoids) components to which the wire is not connected. To pass through intermediary components instead, position the wire at the desired location beside the intermediary component, press `<Shift>` and drag the wire.

4. Click on the desired pin of the second component to finish the connection.

To stop the wiring process at any time, press <Esc>.

To delete a wire, select it and press Delete on your keyboard or right-click on it and choose **Delete** from the pop-up menu that appears.

Combining Automatic and Manual Wiring

You can combine the two methods of wiring when placing a single wire. When in autowiring mode, Multisim assumes you always want to perform automatic wiring until you click somewhere, which "locks" the wire to that point (this is manual wiring). Multisim then continues with automatic wiring, until you click once more—either at a destination pin or wire to complete the connection, or at another interim point on the wire you are placing. This method allows you to use automatic wiring for most connections, and use manual wiring only for difficult, critical or sensitive paths.

Marking Pins for No Connection

You can place NC (no connection) markers on selected component pins to prevent inadvertantly wiring to these pins.

If you attempt to connect a wire to a pin with an NC marker, the connection is prevented and a message appears advising that you cannot connect to a pin marked as "no connection".

Complete the following steps to place an NC marker on a component's pin:

1. Select **Tools»Toggle NC Marker**. The cursor changes to a crosshair.
2. Click the crosshair on the desired pin to place the NC marker ( ).

Complete the following steps to exit the place NC marker mode:

1. Select **Tools»Toggle NC Marker**.

   Or

   Press <Esc>.

   The crosshair on the cursor is replaced with a pointer.

Complete the following steps to place an NC marker on a component's pin via the component's properties dialog:

1. Double-click on the component and click the **Pins** tab.
2. Click in the field in the **NC** column for the desired pin, select **Yes** from the drop-down list that appears and click **OK** to close the dialog box. An NC marker is added to the selected pin on the schematic.

Note You cannot change the NC state of a pin that is already wired to a net. If you try, a warning appears indicating that this is not permitted.

Placing Wires Directly Onto Workspace

For more flexibility during wiring, you can start and end a wire in "mid-air", that is, without attaching it to a component or starting from a previously placed junction.

Complete the following steps to place a wire using the menu:

1. Select **Place»Wire**.

   Or

   Right-click on the workspace and select **Place Schematic»Wire** from the pop-up menu that appears.
2. Click to place a junction on the workspace and then move the mouse to route the wire as desired.
3. Click as desired to lock the wire to a specific point on the workspace.
4. Double-click to place a junction in "mid-air" and end the wire placement.

   Or

   Click to attach the wire to an existing wire or component pin.

Setting Wiring Preferences

Complete the following steps to set how Multisim controls the wiring:

1. Choose **Options»Global Preferences** and select the **General** tab.
2. In the **Wiring** box, enable any or all of the following options:
   - **Automatically connect components when pins are touching** checkbox—Makes an automatic connection if a component pin is touching a wire, another pin or a junction.
   - **Autowire when wiring components** checkbox—Chooses the most efficient path to wire the selected components together. Disabling this option gives you more control over the wire path, as Multisim follows the exact path of your cursor.

Tip While wiring, hold <Shift> to temporarily switch from the selected mode.

   - **Autowire component on move, if number of connections is fewer than** checkbox—This is especially useful for components with large pin counts. Select checkbox and enter the desired

number of connections in the associated listbox. When you move a component with less than this number of connections, Multisim reshapes connected wires to the most efficient path.

When the checkbox is disabled, or a component with more than the set number of connections is moved, the wires are disconnected from the component and remain in their original location.

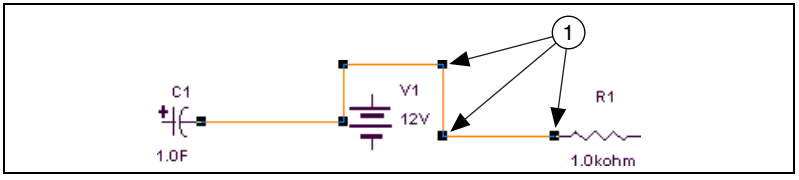
Tip During part movement, hold <Space> to temporarily switch from the selected mode.

3. Enable/disable the **Delete associated wires when deleting component** checkbox. When enabled, wires attached to a component will be deleted when that component is deleted.
4. Click **OK**.

Modifying the Wire Path

Complete the following steps to change the location of a wire once it is placed:

1. Click on the wire. A number of drag points appear on the wire, as shown in (1) in the following figure:



1 Drag Points

2. Click any of these and drag to modify the shape.

   Or, more commonly,

   Move your cursor anywhere on the wire. When your cursor changes to a double arrow, click and drag, in the direction of the arrows, to modify the shape.

You can add or remove a drag point to give you even more control over the wire.

To add or remove drag points, press <Ctrl> on your keyboard and click on the wire at the location where you want the drag point added or removed.

Controlling Wire Color

The default color used for wires is controlled by the **Sheet Properties** dialog box. Refer to the *Sheet Properties - Circuit Tab* section of Chapter 1, *User Interface*, for more information.

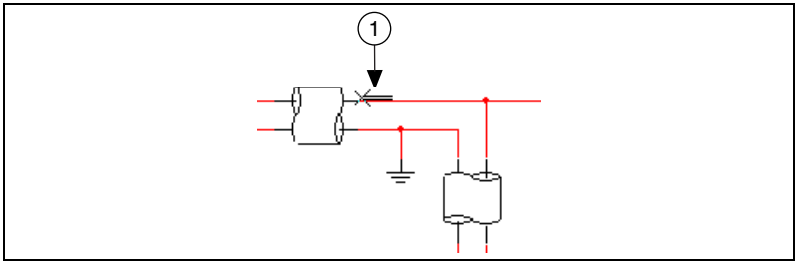
Complete the following steps to change the color of a placed wire, or a wire segment, from its default values:

1. Right-click on the wire and choose **Change Color** or **Segment Color** from the pop-up menu that appears. You are presented with a color palette.
2. Choose a color and click **OK** to apply it to the selected item.

Moving a Wire

Complete the following steps to disconnect a wire and move it to another location in your schematic:

1. Place your cursor at the point where you wish to disconnect the wire. The cursor changes to an “x” with two parallel lines, as shown in the figure below (1).



1 Cursor

2. Click once. The cursor changes to a crosshair.
3. Move the cursor to the pin where you wish to reconnect the wire and click once. The wire is now connected to the new location.

Tip You can create a floating (unconnected) junction and reconnect the wire you are moving to that junction by double-clicking on a blank space on the schematic.

Virtual Wiring

Within a single page of a circuit, a net can be manually renamed to be the same name as another on the same page. In this case, the two nets are merged together. This is called *virtual wiring* and can be used to reduce the complexity of circuits. Except for special reserved nets (for example, Vcc), virtual wiring may not be used across pages or across levels in the hierarchy.

Note You should exercise caution when changing net names, as they are critical to your circuit's connectivity as understood by simulation or PCB layout.

Complete the following steps to make a virtual connection between components:

1. Double-click on the wire. The **Net** dialog box displays. The net name that appears on the schematic is shown in the **Net name** field.
2. Change the **Net name** to match that of the net to which you wish to make the virtual connection. For instance, if you want to connect the selected net to net 21, enter 21 in the **Net name** field.
3. Click **OK**. Multisim prompts you to confirm that you want this duplication.
4. Click **Yes**. Multisim creates a virtual connection between the pins with the same net number.

Note Refer to the *Modifying Net Names* section for more information on the **Net** dialog box.

Manually Adding a Junction

Multisim automatically inserts junctions when you connect one wire to another wire to differentiate them from wires that are crossing but not connected. You can also manually place junctions on the workspace as desired.

Complete the following steps to manually add a junction:

1. Choose **Place»Junction**, or right-click on the workspace and select **Place Schematic»Junction** from the pop-up.
2. Click on the location where you want the junction placed. A junction appears at the selected location.

Note You can place junctions on wires, on the workspace without them being attached to any other circuit element, and directly on the end of component pins. If you place a junction over two intersecting wires, they will be electrically connected.

Complete the following steps to make a connection from a placed junction:

1. Move your cursor close to the junction, until the cursor changes to a crosshair (✦) symbol.
2. Click and drag a wire from the junction to the desired location and click to place.

Rotating/Flipping Placed Components

You can rotate or flip a placed component by using the pop-up menu or selecting the component and using commands from the **Edit** menu. The instructions below describe the pop-up menu method only, but the commands for rotating/flipping placed components that are found in the **Edit** menu are the same. Refer to the *Edit Menu* section of Appendix A, *Menus and Commands*, for more information.

Complete the following steps to rotate a component:

1. Right-click on the component.
2. From the pop-up menu that appears, choose **90 Clockwise** to rotate the component 90 degrees clockwise.

   Or

   Choose **90 CounterCW** to rotate the component 90 degrees counter clockwise.

Note Text associated with the component, such as labels, and values and model information, will be repositioned and rotated as a result of your action. Pin numbers will rotate along with the associated pin. Any wires attached to the component are rerouted automatically to maintain their connections using rubberbanding. If you do not want this to happen, it can be controlled in the **General** tab of the **Preferences** dialog box.

Complete the following steps to flip a component:

1. Right-click on the component.
2. From the pop-up menu that appears, choose **Flip Horizontal** to flip the component horizontally.

   Or

   Choose **Flip Vertical** to flip the component vertically.

Note Text associated with the component, such as labels, model information and values, may be repositioned, but is not flipped. Any wires attached to the component are rerouted automatically.

Finding Components in Your Circuit

You can use the **Find Component** dialog box for larger schematic diagrams where you want to quickly locate a component or net.

Complete the following steps to quickly find a component or net in the workspace:

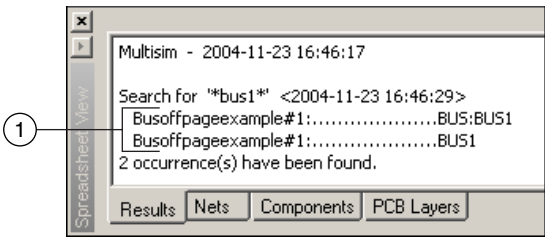
1. Choose **Edit»Find**. The **Find Component** dialog box appears.
2. Enter a string in the **Find what** field that represents all or part of a reference designator (RefDes) or net name as explained below.

   Wildcards are allowed, for example:

   - "V1" finds only the exact string "V1"
   - "*1" finds any string ending with "1"
   - "V*" finds any string starting with "V"
   - "*V*" finds any string containing "V"
   - a "?" anywhere in the string will match exactly one character. For example, "R?" will match "R1", but not "R12".
3. In the **Search for** box, select one of the following radio buttons:
   - **All**—Searches all elements for entered string
   - **Parts**—Searches all parts for entered string
   - **Nets**—Searches all nets for entered string
   - **Off-Page Connectors**—Searches all off-page connectors for entered string
   - **HB/SC Connectors**—Searches all HB/SC connectors for entered string.
4. In the **Search Options** box, select one of the following from the **Search From** drop-down:
   - **Current Sheet**—Search will be conducted on the contents of the current sheet only
   - **Current Design**—Search will be conducted on the contents of the current design only
   - **All Open Sheets**—Search will be conducted on the contents of all open sheets

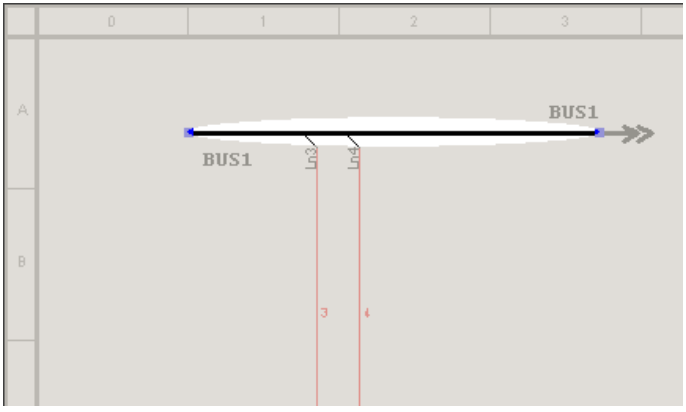
- **All Open Designs**—Search will be conducted on the contents of all open designs.

5. Optionally, select one or both of the following checkboxes in the **Search Options** box:
   - **Match case**—Select if you wish to make the search case sensitive
   - **Match whole word only**—Click if you wish to find only whole words that match the entered string. If *not* selected, search will be for entered string *anywhere*. For example, if "1" is entered, search will yield "V1", R1", "C1", and so on. If you enter a wildcard (*) in the search string, and also select this option, the * will be treated as a character, not a wildcard.
6. Click the **Find** button. The results of your search are displayed in the **Results** tab of the **Spreadsheet View**, as shown in the example in the figure below (1).



1 Search Results

7. Double-click on the desired search result in the **Results** tab. The element is zoomed and selected on the workspace, as shown in the example in the figure below.



8. Click on the workspace to clear the shading from the display.

Note You can also right-click on the desired result and select **Go to** from the pop-up that appears.

Labeling

This section contains the following subjects:

- *Modifying Component Labels and Attributes*
- *Modifying Net Names*
- *Adding a Title Block*
- *Adding Miscellaneous Text*
- *Adding a Comment*
- *Graphic Annotation*
- *Capturing Screen Area*

Modifying Component Labels and Attributes

Complete the following steps to assign a label or change the reference designator (RefDes) of a placed component:

1. Double-click on the component. The component's properties dialog box appears.
2. Click the **Label** tab.
3. Enter or modify the **Label** and/or **RefDes** (which must be composed of letters, numbers and underscores only—no special characters or spaces).
4. Enter or modify the component **Attributes** (which can be any name or value you choose to give them). For example, you could give the component the manufacturer name or a name that is meaningful to you such as "new resistor" or "revised May 15".

   The **Name** field accepts letters, numbers, "+", "-" and "_".

   The **Value** field accepts letters, numbers, "+", "-", "_" and "%".
5. Select the component attributes to display by clicking in the **Show** column and selecting one of **All**, **Value** or **None**. The selected attributes display with the component.
6. Click **OK**.

Note If you assign the same RefDes to more than one component, Multisim warns you that this is not possible. Because all RefDes's must be unique, you must change the RefDes or cancel before you can proceed.

Modifying Net Names

Multisim automatically assigns a net name to each node in the circuit. If desired, you can modify a net name to something more meaningful to the circuit design. For example, you may wish to change a net name to "Output".

Note If you are planning to modify net names for certain "global" reserved nets such as VCC and GND, there are some cautions that apply. Refer to the *Global Nets* section of Chapter 4, *Working with Larger Designs*, for more information.

Complete the following steps to modify a net name:

1. Double-click on the wire. The **Net** dialog box appears.
2. Change the net name as desired.
3. Click **OK**.

Note You should exercise caution when changing net names, as they are critical to your circuit's connectivity as understood by simulation and PCB layout.

Note You are not permitted to change the net name for any wires connected to ground. The net name for these is permanently set to "0". However, you can change the label for the net by double-clicking on any wire attached to ground and entering a name in the **Label** field.

Complete the following steps to lock a net name in position so that it will not move when the wire is moved:

1. Double-click on the net *name* and click **Yes** when prompted.

   Or

   Right-click on the net name and select **Lock/Unlock name position**.

Complete the following steps to unlock a net name that has been locked in place:

1. Double-click on the net *name* and click **Yes** when prompted.

   Or

   Right-click on the net name and de-select **Lock/Unlock name position**.

Adding a Title Block

The **Title Block Editor** is where you create customized title blocks. If desired, a title block can be included on every page of your design.

Various fields in the title block are automatically completed depending upon the context and various document properties. When designing the title block, you can choose a pre-defined field or create your own. You choose appropriate fonts depending upon your language of preference.

Title blocks can include elements such as text, lines, arcs, bezier curves, rectangles, ovals, arcs, and bitmaps.

Complete the following steps to add a title block to your circuit:

1. Choose **Place»Title Block**. A standard **Open** dialog box appears. If necessary, navigate to the `titleblocks` folder.
2. Select the desired title block template and click **Open**. The selected title block appears attached to your cursor. Drag and drop it to the desired location, typically the lower-right corner of the page.
3. You can also move the placed title block by right-clicking on it and selecting one of:
   - **Move to»Top Left**—Places the title block in the top-left corner of the workspace.
   - **Move to»Top Right**—Places the title block in the top-right corner of the workspace.
   - **Move to»Bottom Left**—Places the title block in the bottom-left corner of the workspace.
   - **Move to»Bottom Right**—Places the title block in the bottom-right corner of the workspace.

To add a title block with a new format, refer to the *Title Block Editor* section of Chapter 3, *Schematic Capture - Advanced Functions*, and then place the new title block on the circuit following the procedure above.

Entering the Title Block Contents

Complete the following steps to edit the contents of the title block:

1. Right-click on the title block and select **Properties** from the pop-up that appears.

   Or

   Double-click on the title block.

   The **Title Block** dialog box appears.

2. The information that displays is described in this step. Make edits as desired and click **OK**.

Note If a field in the **Title Block** dialog box contains information, and that information does not appear in your title block, it is because the field that contains that information was not placed in the title block. If this occurs, right-click on the title block, select **Title Block Editor**, and place the required field in the title block. Refer to the *Placing Fields* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.

- **Title**—Title of the circuit. Defaults to the circuit's filename. Corresponds to the `#TITLE` field that is placed using the **Title Block Editor**.
- **Description**—A description for your project. Defaults to **Project 1**. Corresponds to the `#DSCRPT` field that is placed using the **Title Block Editor**.
- **Designed by**—Designer's name. Corresponds to the `#DESIGNED` field that is placed using the **Title Block Editor**.
- **Document No.**—Document number. Defaults to **0001**. Corresponds to the `#DOC_N` field that is placed using the **Title Block Editor**.
- **Revision**—Revision number of the circuit. Corresponds to the `#REV` field that is placed using the **Title Block Editor**.
- **Checked by**—Name of person checking the circuit. Corresponds to the `#CHECKED` field that is placed using the **Title Block Editor**.
- **Date**—Defaults to the date that the circuit was created. Corresponds to the `#DATE` field that is placed using the **Title Block Editor**.
- **Size**—Size of the sheet, for example, "A". Corresponds to the `#FMT` field that is placed using the **Title Block Editor**.
- **Approved by**—Name of person approving the circuit. Corresponds to the `#APPROVED` field that is placed using the **Title Block Editor**.
- **Sheet**—The number of the current sheet and the total number of sheets. For example, Sheet 2 of 3. Corresponds to the `#SN` and `#TSN` fields that are placed using the **Title Block Editor**.
- **Custom Field 1**—Information entered here will be placed in Custom Field 1. Corresponds to the `#CUSTOM_1` field that is placed using the **Title Block Editor**.

- **Custom Field 2**—Information entered here will be placed in Custom Field 2. Corresponds to the `#CUSTOM_2` field that is placed using the **Title Block Editor**.
- **Custom Field 3**—Information entered here will be placed in Custom Field 3. Corresponds to the `#CUSTOM_3` field that is placed using the **Title Block Editor**.
- **Custom Field 4**—Information entered here will be placed in Custom Field 4. Corresponds to the `#CUSTOM_4` field that is placed using the **Title Block Editor**.
- **Custom Field 5**—Information entered here will be placed in Custom Field 5. Corresponds to the `#CUSTOM_5` field that is placed using the **Title Block Editor**.

Caution The width of a *field* as displayed in the **Title Block Editor** is not the same as the actual *text* that is placed in that field in the title block. The text in the title block will be as wide as the amount of text. It is also dependant on the font size. So if two fields are placed closed to each other they may overlap.

You can also place the following special symbols in the **Title Block** dialog box. When you return to the workspace, the symbol will be replaced by text as described below:

- **&p**—Page number
- **&P**—Total number of pages
- **&d**—Date
- **&t**—Time
- **&s**—Page name
- **&j**—Project name
- **&&**—Ampersand.

Adding Miscellaneous Text

Multisim allows you to add text to a circuit, for example to label a particular part of a circuit.

Complete the following steps to add text:

1. Choose **Place»Text** or right-click on the workspace and select **Place Graphic»Text** from the pop-up.
2. Click on the location where you want the text placed. A text box with a blinking cursor appears.

3. Type the text. The text box correctly sizes when you finish typing and click elsewhere in the workspace.
4. Click elsewhere on the circuit window to stop adding text.

Complete the following steps to delete text:

1. Right-click on the text box.
2. Choose **Delete** from the pop-up menu that appears.

   Or

1. Select the text and press Delete on your keyboard.

Complete the following steps to change the color of text:

1. Right-click on the text box.
2. Choose **Pen Color** from the pop-up menu that appears, and choose the desired color.

   Refer to the *Pop-up From a Selected Text Block or Graphic* section of Chapter 1, *User Interface*, for more information.

Complete the following steps to change the font options for the text:

1. Right-click on the text box.
2. Choose **Font** from the pop-up menu that appears, and choose the desired font options.

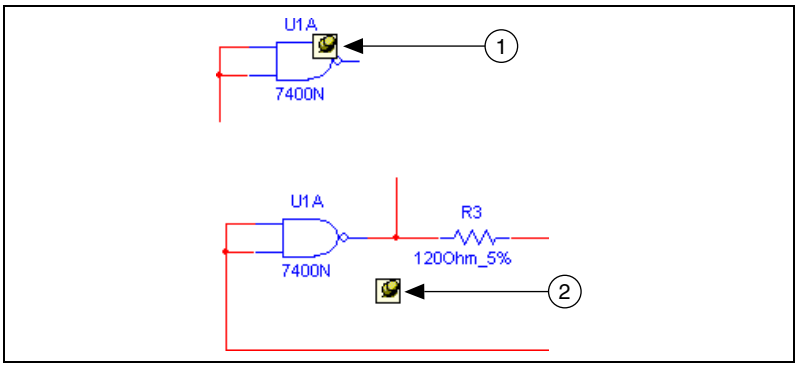
Adding a Comment

Adding a comment permits “redlining”, which can be used to show engineering change orders, to facilitate collaborative work among team members, or to allow background information to be attached to a design.

You can “pin” a comment to the workspace, or directly to a component. When a component with an attached comment is moved, the comment also moves.

Complete the following steps to pin a comment to a component or the workspace:

1. Select **Place»Comment**.
2. Move the cursor to the desired location and click to place the comment.



1 Comment Pinned to Component 2 Comment on Workspace

The figure above shows a comment “pinned” to a component (1); if the component is moved, the comment moves with it. A comment can also be placed on the workspace, as shown in the figure above (2). It does not move when components are moved.

Complete the following steps to enter text in the placed comment:

1. Double-click on the placed comment and the select the **Display** tab. The **Comment Properties** dialog box appears. The layer on which the comment appears displays in the **Drawing Layer** list. Change this if desired. Refer to the *Visibility Tab* section of Chapter 1, *User Interface*, for more information.
2. Type the desired text in the field at the bottom of the dialog. If you wish to show the note’s contents, enable the **Show popup window** checkbox.
3. Optionally, set the **Background** and **Text** colors in the **Color** box.

Note “Tooltip” refers to the text that appears attached to the cursor if you hover it above a tool button.

4. In the **Size** box, enter the **Width** and **Height**, or enable **Auto-Resize** to have the info box automatically resize to show all content.
5. Optionally, click on the **Font** tab to change the comment’s font.
6. Click **OK**. If you checked **Show popup window** in the **Display** tab, the comment displays. Otherwise, the **Comment** icon displays.

To enter text without using the **Comment Properties** dialog box, right-click on the placed comment, select **Edit Comment** from the pop-up that appears and type the desired text.

Complete the following steps to display a hidden comment:

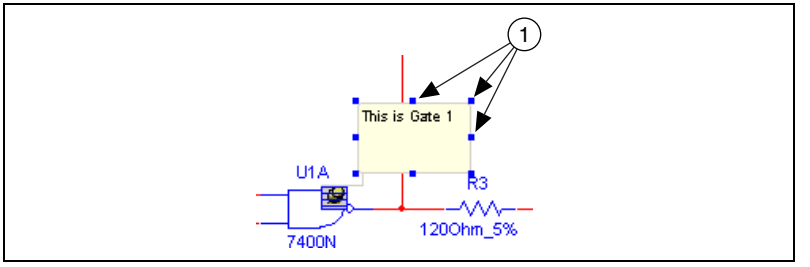
1. Right-click on the desired **Comment** icon.
2. Select **Show Comment/Probe** from the pop-up that appears.

Complete the following steps to see the contents of a hidden comment:

1. Hover the cursor over the comment.
2. Move the cursor to hide the comment.

Complete the following steps to change the size of a displayed comment:

1. Select its textbox.
2. Drag the handles that appear, as in the example (1) in the figure below.



1 Drag Handles

Note The size can also be changed from the **Display** tab of the **Comment Properties** dialog box.

Graphic Annotation

Use **Graphic Annotation** to add the following graphic elements to your workspace:

- Line
- Multiline
- Rectangle
- Ellipse
- Arc
- Polygon
- Picture
- Comment.

Complete the following steps to add a graphic element:

1. If the **Graphic Annotation** toolbar is not already showing, select **View»Toolbars»Graphic Annotation**.

   Or

   Right-click in the menu area and select **Graphic Annotation** from the pop-up that appears.

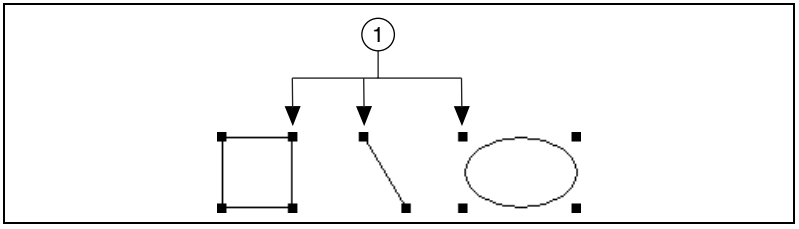
2. Click on the button in the **Graphic Annotation** toolbar for the desired graphic element and follow the directions for that element from the table below.

Button	Description
	Click on this button to place text on the workspace. Then click on the workspace in the location where you wish to place the text and type in the desired text. When finished, click anywhere on the workspace. The text box automatically sizes to display your text.
	Click on this button to draw a line. The cursor changes to a crosshair. Place the crosshair where you wish to start the line and click and hold the mouse button. Drag the crosshair to where you wish to end the line and release the mouse button. (To add an arrowhead, right-click on the placed line and select **Arrow** from the pop-up).
	Click on this button to draw a multiline, which consists of multiple connected line segments. The cursor changes to a crosshair. Place the crosshair where you wish to start the multiline and click the mouse button once. Move the crosshair to where you wish to end the current segment of the multiline and click the button once. Repeat until all segments of the multiline have been drawn. When you have drawn the last segment of the multiline, double-click the mouse button. (To add an arrowhead, right-click on the placed line and select **Arrow** from the pop-up).
	Click on this button to draw a rectangle. The cursor changes to a crosshair. Place the crosshair where you wish to start the rectangle and click and hold the mouse button. Drag the crosshair to where you wish to end the rectangle and release the mouse button.

Button	Description
	Click on this button to draw an ellipse. The cursor changes to a crosshair. Place the crosshair where you wish to place the center of the ellipse and click and hold the mouse button. Drag the crosshair to where you wish to end the ellipse and release the mouse button.
	Click on this button to draw an arc. The cursor changes to a crosshair. Place the crosshair where you wish to place the center of the arc and click and hold the mouse button. Drag the crosshair to where you wish to end the arc and release the mouse button. The arc will appear as an ellipse while it is being drawn, but when released, the right side only of the ellipse will be shown, thereby giving the arc.
	Click on this button to draw a polygon. The cursor changes to a crosshair. Place the crosshair where you wish to start the polygon and click the mouse button once. Move the crosshair to where you wish to end the current segment of the polygon and click the button once. Repeat until all segments of the polygon have been drawn. When you have drawn the last segment of the polygon, double-click the mouse button.
	Click on this button to place a picture on the workspace. A dialog opens from where you can select the desired bitmap image. You can place either a `.bmp` or `.dib` file.
	Click on this button to place a comment on the workspace. For details, see Adding a Comment.

Complete the following steps to change the size of placed graphic elements:

1. Select the graphic.
2. Click and drag the drag points that appear, as in the example (1) shown in the figure below.



1 Drag Points

Note Placed graphics can also be manipulated using the pop-up menu. Refer to the *Pop-up From a Selected Text Block or Graphic* section of Chapter 1, *User Interface*, for more information.

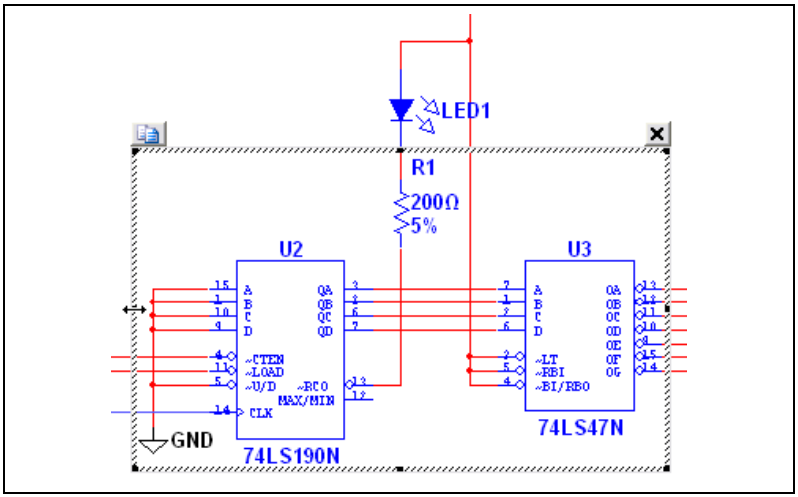
Capturing Screen Area

You can capture an area of the screen and then manipulate the image as you would any other screen capture contained in the system clipboard. For example, you can paste it into the **Circuit Description Box**. Refer to the *Circuit Description Box* section for more information.

Complete the following steps to copy a section of your screen to the clipboard:

1. Select **Tools»Capture Screen Area**. A selection frame appears on your workspace.
2. Optionally, to move the frame to a different location:
   - Move your cursor to the border of the frame. A crosshair is added to the cursor, indicating that the selection frame can be moved to a different location.
   - Drag the selection frame to the desired location.

3. Optionally, to re-size the selection frame:
   - Move the cursor to one of the sizing handles.



   - Drag the cursor to re-size the selection frame.

4. Click the **copy** button at the top left corner of the selection frame. The image inside the selection frame is copied to the system clipboard.
5. Click the **x** at the top right corner of the selection frame to close it.

Circuit Description Box

In addition to adding text to a particular portion of a circuit, you can add general descriptions to your circuit using the **Circuit Description Box**. You can also place bitmaps, sound and video in the **Circuit Description Box**.

The contents of the **Circuit Description Box** are *viewed* in the top pane of the **Circuit Description Box** window (select **View»Circuit Description Box**). To *edit* the contents of the **Circuit Description Box**, select **Tools»Description Box Editor**.

Note The bottom pane of the **Circuit Description Box** contains any questions that you have entered using Multisim's **Forms** functionality. Refer to the *Linking a Form to a Circuit* section for more information.

Complete the following steps to add or edit a description:

1. Choose **Tools»Description Box Editor**. The **Edit Description** window appears.
2. Enter text by typing directly into the window, or choose **Insert» Insert Object** to place a bitmap, etc.

Note Changes made in the **Edit Description** window are not reflected in the **Circuit Description Box** (selected by **View»Circuit Description Box**) until you exit the **Edit Description** window.

3. Use the **Description Edit Bar** to edit the contents of the **Circuit Description Box** as needed. Refer to the *Description Edit Bar* section for more information.
4. When you are finished entering text, select **File»Close**. The **Edit Description** window closes and you are returned to the main Multisim workspace.

To print your description, from the **Edit Description** window, click the **Print** button.

Formatting the Circuit Description Box

You can format the contents of the **Circuit Description Box** as listed below:

- *Paragraph Dialog Box*—Use to enter paragraph formatting information.
- *Tabs Dialog Box*—Use to enter tab settings.
- *Date and Time Dialog Box*—Use to place a formatted date and/or time.
- *Properties Dialog Box*—Use to set measurement units and text wrapping settings.
- *Insert Object Dialog Box*—Use to select the type of object for insertion.

Formatting Circuit Description Box Text

You can format currently-selected text in the **Circuit Description Box** by using the following commands from the **Description Edit Bar**:

- **Font** button—Click to select the desired font settings.
- **Bold** button—Click to make the selection bold.
- **Italic** button—Click to italicize the selection.
- **Underline** button—Click to underline the selection.

Note The above commands are also available by selecting **Format»Font**.

- **Color** button—Click to display a palette where you pick a color for the text. This palette will also display if you select **Format»Font Color**.
- **Left Justification** button—Click to align the selected paragraph(s) along the left margin. You can also select **Format»Align Left** to perform this command.
- **Center Justification** button—Click to center-align the selected paragraph(s). You can also select **Format»Align Center** to perform this command.
- **Right Justification** button—Click to align the selected paragraph(s) along the right margin. You can also select **Format»Align Right** to perform this command.
- **Insert Bullet** button—Click to insert a bullet at the beginning of the selected paragraph(s). You can also select **Format»Insert a Bullet** to perform this command.

Note Refer to the *Description Edit Bar* section for more information.

Paragraph Dialog Box

Use the **Paragraph** dialog box to enter paragraph formatting information for the **Circuit Description Box**.

Complete the following steps to enter paragraph formatting information for the **Circuit Description Box**:

1. Open the **Edit Description** window as described in the *Circuit Description Box* section.
2. Select **Format»Paragraph** to display the **Paragraph** dialog box.

3. In the **Indentation** box, enter the measurements in the following fields as desired:
   - **Left**—The distance the left side of the paragraph is indented from the left margin.
   - **Right**—The distance the right side of the paragraph is indented from the right margin.
   - **First Line**—The distance the first line of the paragraph is indented from the left margin.
4. In the **Alignment** drop-down list, select whether the paragraph is left-, right-, or center-aligned.
5. Click **OK**.

Tabs Dialog Box

Use the **Tabs** dialog box to enter tab settings for the **Circuit Description Box**.

Complete the following steps to enter tab settings for the **Circuit Description Box**:

1. Open the **Edit Description** window as described in the *Circuit Description Box* section.
2. Select **Format»Tab** to display the **Tabs** dialog box.

3. Enter the desired position for the tab (for example 1.25”) in the **Tab Stop Position** field and click **Set**.

Note You can also click in the top ruler bar to place a tab.

Complete the following steps to remove a tab setting:

1. Highlight the desired tab setting and click **Clear**.
2. To clear all tabs, click **Clear All**.

Date and Time Dialog Box

Use the **Date and Time** dialog box to enter a formatted date and/or time in the **Circuit Description Box**.

Complete the following steps to place a date and/or time in the **Circuit Description Box**:

1. Open the **Edit Description** window as described in the *Circuit Description Box* section.

2. Click at the location where you wish to place the date and/or time.
3. Select **Insert»Insert Date** to display the **Date and Time** dialog box.
4. Select the desired date/time format from the **Available formats** list and click **OK** to place the selection.

Properties Dialog Box

Use the **Properties** dialog box to select the measurement units and text wrapping settings used in the **Circuit Description Box**.

Complete the following steps to set the measurement units and text wrapping settings:

1. Open the **Edit Description** window as described in the *Circuit Description Box* section.
2. Select **Options»Rich Edit Options** to display the **Properties** dialog box and click the **Options** tab.
3. In the **Measurement units** box, select one of **Inches**; **Centimeters**; **Points**; **Picas**.
4. Optionally, enable **Automatic Word Selection** to select one word at a time when you drag the cursor. If you wish to select one character at a time, clear this checkbox.
5. Click the **Rich Text** tab and select one of:
   - **No wrap**—Typed text will be on one line until you press the ENTER key.
   - **Wrap to window**—Typed text will go to the next line when it reaches the edge of the window.
   - **Wrap to ruler**—Typed text will go to the next line when it reaches the page margin.
6. Click **OK**.

Insert Object Dialog Box

Use the **Insert Object** dialog box to select an object, such as a bitmap or chart, to insert in the **Circuit Description Box**.

Complete the following steps to insert an object:

1. Open the **Edit Description** window as described in the *Circuit Description Box* section.
2. Select **Insert»Insert Object** to display the **Insert Object** dialog box.
3. To create a new object, enable **Create New**.

4. Select the desired object from the **Object Type** list.
5. Optionally, enable **Display As Icon** to view an icon representing the file.
6. Click **OK**.

Complete the following steps to create an object from an existing file:

1. Enable the **Create from File** button.
2. Enter the desired filepath and name in the **File** field, or click **Browse** and navigate to the desired file from the **Browse** dialog box that appears.
3. Optionally, enable **Link** to link the object to the original file. Any updates to the original file are reflected in the object.

Scrolling with Events During Simulation

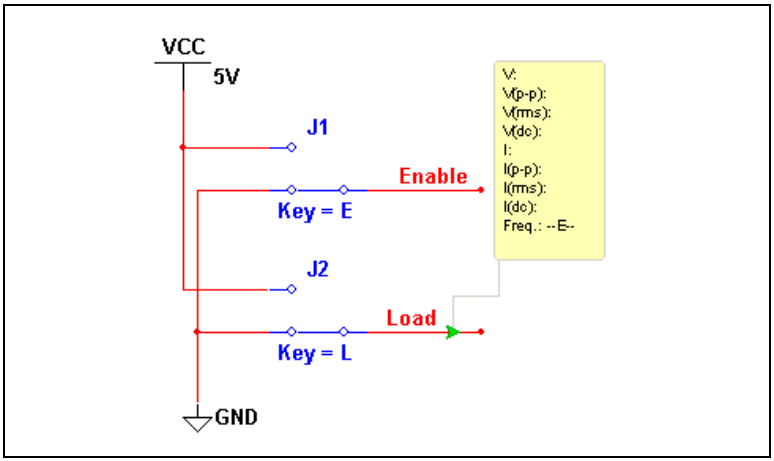
Along with descriptive text and graphics, the **Circuit Description Box** lets you synchronize scrolling and display of text, graphics or video with events in the simulation. You do this by placing measurement probes in your design and setting the probe properties with the events that you would like to have synchronized with the scrolling or display of text, graphics or video.

Scrolling Text During Simulation

Complete the following steps to make text scroll automatically during a simulation:

1. Enter text in the **Circuit Description Box** as desired.
2. Select **Simulate»Instruments»Measurement Probe** and click to place a probe at the desired point, as in the figure below. The simulation must *not* be running.

This is the point where the desired event occurs. For example, when the voltage is equal to 5 V.



Note Refer to the *Measurement Probe* section of Chapter 9, *Instruments*, for more information on the probe.

3. Double-click on the placed probe to display the **Probe Properties** dialog box, and click on the **Description Box** tab.
4. Click **New**. The blinking text cursor moves to the **Condition(s)** field.
5. Click on the button to the right of the **Condition(s)** field and build the equation from the pop-up that appears. In this case we are entering the condition "V=5", meaning the condition required to scroll the text will be achieved when the voltage at the probe equals 5 volts.
6. From the **Action** drop-down list, select **Start Scrolling**.
7. In the **Parameter** field, type the scroll speed, for example, "100" (this equals a scroll rate of 100 pixels per 10 seconds).
8. Click **Apply**.
9. Click **OK** to close the dialog box.

   When you run the simulation, the text will scroll when the voltage at the probe equals 5 V.

Note To disable a trigger, highlight it in the **Description Box Triggers** area and deselect the **Enabled** checkbox.

10. Click **Run/Resume Simulation** in the **Simulation** toolbar. The text in the **Circuit Description Box** will scroll when the conditions set in the **Description Box** tab of the **Probe Properties** dialog box are met.

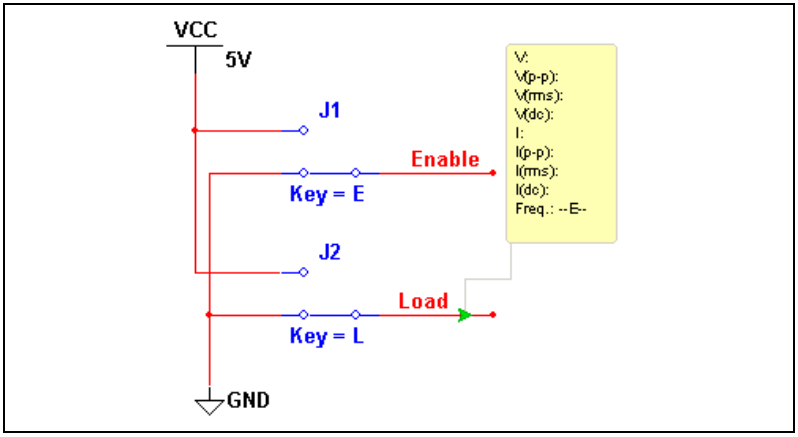
Note Remember to leave the **Circuit Description Box** open when you click **Simulate**.

Playing a Video Clip

Complete the following steps to play a video clip during simulation:

1. Open the **Edit Description** window by selecting **Tools»Description Box Editor**.
2. Click at the point where you wish to place the video clip.
3. Select **Insert»Insert Object** and select **Create from File**.
4. If you wish to have changes to the original file reflected in Multisim, enable **Link**.
5. Click **Browse** and navigate to the desired video clip.
6. Click **OK** to place the clip.
7. Click once on the clip to select it and select **Insert»Insert Label**. Enter the name of the label in the **Label Name** field of the **Description Label** dialog that appears and click **OK**. The clip's filename (in this case, "Filmclip.mpg") appears enclosed in rails as shown in the figure below.

8. Select **Simulate»Instruments»Measurement Probe** and click to place a probe at the desired point, as in the example in the figure below. The simulation must *not* be running at this point.



Note This point is where the desired condition will occur. For example, the voltage becomes equal to 5 V.

9. Double-click on the placed probe to display the **Probe Properties** dialog box, and click on the **Description Box** tab.
10. Click **New**. The blinking text cursor moves to the **Condition(s)** field.
11. Click on the button to the right of the **Condition(s)** field and build the equation from the pop-up that apppears. In this case we are entering the condition "V=5", meaning the condition required to run the clip will be achieved when the voltage at the probe is 5 volts.
12. From the **Action** drop-down list, select **Play Media Clip**.
13. In the **Parameter** field, type the name of the label that you placed around the video clip in the **Circuit Description Box**, (as described in the *Scrolling with Events During Simulation* section). In this case, we used Label 2.
14. Click **Accept**.
15. Click **OK** to close the dialog box.

    When you run the simulation, the video clip will play when the voltage at the probe equals 5 V.

16. Click **Run/Resume Simulation** in the **Simulation** toolbar. The video clip will run when the parameters set in the **Description Box** tab of the **Probe Properties** dialog box are met.

Note Remember to leave the **Circuit Description Box** open when you click **Simulate**. You can also run audio clips during simulation using the above procedure.

Description Label Dialog Box

You can insert labels in the **Circuit Description Box** that mark points where a certain action is to occur during simulation. For example, you might want to jump to a specific point in the text when the voltage at a specific node in the circuit drops below a pre-set value. Or, you may wish to run a film clip when another parameter has been met. The specific point in the text, and the film clip, are both marked with labels.

These actions are set up in the **Description Box** tab of the **Probe Properties** dialog box.

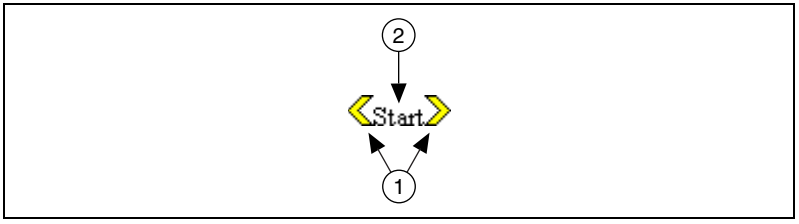
Use the **Description Label** dialog box to create a label to insert in the **Circuit Description Box**.

Complete the following steps to insert a label:

1. Open the **Edit Description** window as described in the *Circuit Description Box* section.
2. Select a block of text or an object that you wish to label. (This will be the *content* of the label.)
3. Select **Insert»Insert Label** to display the **Insert Label** dialog box.
4. Enter a *name* for the label in the **Label Name** field.
5. If you wish the content of the label to display in the **Circuit Description Box**, enable the **Show content in description bar** checkbox.

Note The content of the label is the text or object that you selected in step 2.

6. Click **OK** to accept your selections and close the dialog box. The label contents appear as show in the example in the figure below (2), enclosed by upper and lower rails (1).



1	Rails Indicate Label	2	Label Contents

Edit Labels Dialog Box

Complete the following steps to edit existing labels in the **Circuit Description Box**:

1. Open the **Edit Description** window as described in the *Circuit Description Box* section.
2. Select **Edit»Labels** to display the **Edit Labels** dialog box.

To show/hide the contents of labels in the **Circuit Description Box**, enable/disable checkboxes as desired.

To delete a label, highlight the desired label and click **Remove**.

To rename a label, highlight the desired label and click **Rename**.

Other Actions

You can set up the following actions in the **Description Box** tab of the **Probe Properties** dialog box:

- **Jump to Label**—When the conditions entered in the **Condition** field are met (for example, V=5), the text in the **Circuit Description Box** will jump to the label that you entered in the **Parameter** field.
- **Pause Simulation**—When the conditions entered in the **Condition** field are met (for example, V=5), the simulation will pause for the amount of time (in seconds) that you entered in the **Parameter** field.
- **Stop Scrolling**—When the conditions entered in the **Condition** field are met (for example, V=5), scrolling of the text in the **Circuit Description Box** will stop.

Description Edit Bar

The **Description Edit Bar** contains the buttons described in the table below.

Button	Description
	Add Question Link button. Inserts a question link at the current cursor position. Refer to the *Linking a Form to a Circuit* section for more information.
	Insert Date and Time button. Displays the **Date and Time** dialog box, where you select the format for the date and time. Refer to the *Date and Time Dialog Box* section for more information.
	View and Select Options button. Displays the **Options** dialog box, where you select measurement units and text wrapping settings. Refer to the *Properties Dialog Box* section for more information.
	Insert Embedded Object button. Displays the **Insert Object** dialog box, where you select the type of object to insert. Refer to the *Insert Object Dialog Box* section for more information.
	Create New Label button. Displays the **Description Label** dialog box, where you enter the name of the new label to be inserted. Refer to the *Description Label Dialog Box* section for more information.

Button	Description
	Edit Labels button. Displays the **Edit Labels** dialog box. Refer to the *Edit Labels Dialog Box* section for more information.
	Bold button. Makes the selection bold.
	Italic button. Makes the selection italic.
	Underline button. Underlines the selection.
	Left Justification button. Aligns the selected paragraph(s) along the left margin.
	Center Justification button. Center-aligns the selected paragraph(s).
	Right Justification button. Aligns the selected paragraph(s) along the right margin.
	Font button. Changes the font, size and color of the selected text.
	Color button. Displays a color palette where you pick a color for the currently-selected text.
	Paragraph Format button. Displays the **Paragraph** dialog box, where you enter paragraph formatting information. Refer to the *Paragraph Dialog Box* section for more information.
	Insert Bullet button. Inserts a bullet at the beginning of the selected paragraph(s).
	Insert Tabs button. Displays the **Tabs** dialog box, where you enter tab formatting information. Refer to the *Tabs Dialog Box* section for more information.

Linking a Form to a Circuit

You can use Multisim's form functionality to send circuits for approval, design reviews, or anywhere feedback on a design is needed. Once the form has been completed, the circuit file, including the completed form, can be returned to the originator via email.

Education edition users will find this particularly useful for assignments and tests that are given to students to be completed remotely. Once the questions have been answered, the circuit file, including the completed form, can be emailed to the instructor at the click of a button.

A form can include any of the following types of questions:

- **Multiple Choice**—The correct response is selected from two or more possible responses.
- **True/False**—A statement is made on the form and either **True** or **False** is selected as the response.
- **Data Entry**—A field for limited data entry is provided.
- **Free Form**—A field for more extensive data entry is provided.

You can also insert links in the **Circuit Description Box** that will, when double-clicked, take the person completing the form directly to a linked question. This is useful where clarification or further explanation of a specific question may require more text than you want to put on the form. Refer to the *Linking to Questions* section for more information.

The following sections discuss how to create a form, how to link to the form from the **Circuit Description Box**, how to set form submission options, and how the recipient should complete and submit the form.

Creating Forms

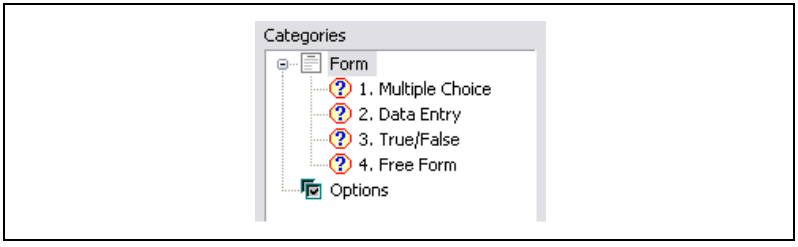
When reviewing these instructions, please note that the dialog box where you enter the questions is called **Edit Questions** in the Education Edition of Multisim. To access this dialog, select **Edit»Forms/Questions**.

Complete the following steps to create a form:

1. Select **Edit»Forms/Questions**. The **Edit Form** dialog box displays.
    - **Title**—Text entered here appears as a title when the form is viewed.

- **Instruction**—Enter instructions as desired in this field. Text entered here appears above the first question when the form is viewed.
- **User Profile**—Each line entered here appears as a separate line with a user-editable field when the questions are viewed from the form. You can add, edit, or delete items in this area as desired.

2. Click on the **Add a question** button and select the desired question type from the pop-up that appears. The **Edit Form** dialog box changes to reflect your selection.
3. Enter the question based on the following:
   - **Multiple Choice**—Enter a question in the **Question** field and enter possible responses in the **Candidate Answers** fields.
   - **True/False**—Enter a question requiring a true or false answer in the **Question** field.
   - **Data Entry**—Enter a question or instruction in the **Question** field.
   - **Free Form**—Enter a question or instruction in the **Free Form** field.
4. As you proceed, the question types appear in the **Form** tree of the **Edit Form** dialog box as shown in the figure below.



Click on a specific question in the tree to view its contents in the dialog box.

Click on the **Remove Selected Question** button to remove it completely.

Use the up and down arrows to change the position of the selected question in the tree. The sequence of the questions in the tree will be reflected when you view the completed form in lower pane of the **Circuit Description Box**.

5. Once you have entered the desired questions, click **OK** to close the **Edit Form** dialog box.

6. Select **View»Circuit Description Box**. The questions are displayed in the bottom pane of the **Circuit Description Box**.
7. Save the Multisim circuit file, which now includes the form. The completed circuit file can be electronically sent (for example, via email) to the desired recipients.

Linking to Questions

You can insert links in the **Circuit Description Box** that will, when double-clicked, take the person completing the form directly to the linked question.

Complete the following steps to insert links in the **Circuit Description Box** to specific questions in a form:

1. Create the form as detailed in the *Creating Forms* section.
2. Select **Tools»Description Box Editor**.
3. Enter text, graphics, etc. in the usual fashion. Refer to the *Circuit Description Box* section for more information.

4. Place the cursor where you wish to insert a link and select **Insert»Insert Question Link**. The **Select a Form** dialog box appears, listing all questions that are on the current form.
5. Highlight the question that you wish to link to and click **OK**. The link is placed at the cursor's positon.
6. Insert other links as desired and close the **Description Box Editor**. Your changes, including the links, appear in the **Circuit Description Box**.
7. Double-click on a link in the **Circuit Description Box** to jump to the linked question in the lower pane of the **Circuit Description Box**.

Note Refer to the *Completing Forms* section for more information.

Setting Form Submission Options

Form submission options are usually set by the person that creates the form, before it is sent for completion.

When reviewing these instructions, please note that the dialog box where you enter the questions is called **Edit Questions** in the Education Edition of Multisim. To access this dialog, select **Edit»Questions**.

Complete the following steps to set options for submitting completed questions:

1. Select **Edit»Forms/Questions**. The **Edit Form** dialog box displays.
2. Click **Options** in the **Categories** area and complete the following:
   - **Email the circuit file to**—Enable checkbox and enter the desired email address.
   - **Subject**—Enter text to be placed in the email's subject line.
   - **Body**—Enter text to be placed in the body of the email message.
   - **Display the message**—Enable checkbox and enter text that you wish to appear in the message that appears on the Multisim workspace when you click **Submit**.
3. Click **OK** to close the **Edit Form** dialog box.

Note Refer to the *Creating Forms* and the *Completing Forms* sections for more information.

Completing Forms

Complete and submit the form from a Multisim circuit that you received as explained below. Questions are answered directly from the **Circuit Description Box**.

Complete the following steps to answer questions:

1. Select **View»Circuit Description Box**.
2. Enter the default information in the fields found at the top of the form in the lower pane of the **Circuit Description Box** (for example, **Name**, **Date**). These fields may vary from circuit to circuit.
3. Complete the questions by selecting the desired response for multiple choice and true/false questions, or typing in answers to data entry and free form questions.

Tip If the **Circuit Description Box** contains links to questions , you can double-click on a link to jump directly to the associated question.

4. Submit the completed questions as described below:
   - To submit the completed questions on paper, click **Print**. A standard **Print** dialog appears. Enter the desired settings and click **Print**.
   - To submit the completed questions by email, click **Submit**. The circuit file, including the completed questions will be attached to

an email that is addressed as set up in the **On Submit** options. Refer to the *Setting Form Submission Options* section for more information.

Send this email in the usual manner.

Printing the Circuit

Multisim allows you to control specific aspects of your printing, including:

- whether to output in color or black and white.
- which pages of a design to print.
- whether to include the background in the printed output.
- page margins for printing.
- scaling of the circuit's image to fit the printed output.

Complete the following steps to set the printing environment for circuits:

1. Select **File»Print Options»Print Circuit Setup**.
2. Set the **Page Margins**, **Zooms** and **Page Orientation** options as desired.
3. Set the **Output Options** as desired:
   - **In Black/White**—Prints the circuit in black and white (for non-color printers). When disabled, colored components print in shades of grey.
   - **Instruments**—Prints the circuit and faces of the instruments used in the circuit on separate sheets.
   - **Background**—Includes the background in printed output. Use for color printers or white on black output. This option is disabled if **In Black/White** is selected.
   - **Current Circuit**—Prints the window that is currently active on the workspace.
   - **Current and Subcircuits**—Prints the currently active window and any subcircuits or hierarchical blocks that it contains.
   - **Entire Design**—Prints all circuits, subcircuits, hierarchical blocks and multi-pages from the design which includes the currently active window. Refer to the *Hierarchical Design* section of Chapter 4, *Working with Larger Designs*, for more information about hierarchical blocks and subcircuits.

4. Click **OK** to set the printing environment for the current circuit, or click **Set As Default** to set the printing environment for all circuits.

To preview your file before printing, choose **File»Print Preview**. The circuit appears in a preview window where you can zoom in, move from page to page, and send the circuit to the printer using the provided tool buttons.

To print the circuit file, choose **File»Print**. We recommend that you set your print options first.

3

Schematic Capture - Advanced Functions

This chapter describes the advanced functions involved in creating a circuit in Multisim.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Placed Component Properties

Each component placed on the circuit window has a set of properties that control certain aspects of it beyond those stored in the Multisim database. These properties affect only the placed component, not other instances of that component in other circuits or other locations in this circuit. Depending on the type of component, these properties determine some or all of the following:

- the identifying information and labels about the placed component to be displayed on the circuit window. Refer to the *Modifying Component Labels and Attributes* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- the model of the placed component.
- for some components, how the placed component will be used in analyses.
- the faults to be used for the placed component.
- the component's value or model and footprint.
- user fields.

Displaying Identifying Information about a Placed Component

The settings in the **Circuit** tab of the **Sheet Properties** dialog box determine which identifying information is displayed on your circuit. Refer to the *Sheet Properties - Circuit Tab* section of Chapter 1, *User Interface*, for more information.

You can also override these settings for an individual placed component, as described here.

Complete the following steps to set the identifying information to be displayed for a placed component:

1. Double-click on the component. A properties dialog box for the selected component appears.
2. Click the **Display** tab.
3. Disable **Use Schematic Global Setting**. The "**Show**" options become available.

Note When this option is enabled, the types of identifying information displayed for the selected component are controlled by the circuit's settings.

4. Enable the identifying information you want displayed for this component, and disable the identifying information you do not want displayed for this component. For example, if you wish to hide the component's reference designator, disable **Show RefDes**.
5. Click **OK** to save your settings.

Viewing a Placed Component's Value/Model

The **Value** tab of the "properties" dialog box for a component shows the value/model being used for the placed component. Depending on the component, the contents of the **Value** tab will differ when you double-click on a placed component.

Real Components

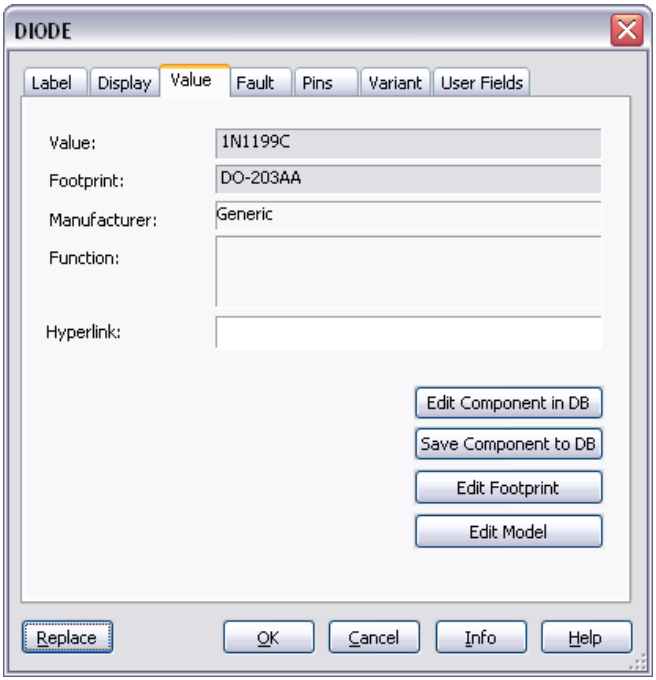
In one sense of the word, all components found in Multisim are virtual. They are virtual representations of components like diodes and transistors. Wired together and simulated in Multisim, they will give you an idea of how such a circuit will function when prototyped.

However, when a "real" component is referred to in this manual, it corresponds to a real, purchasable component with values that include footprints and package types for use in PCB layout programs, like Ultiboard.

"Virtual" components, however, give you a means to experiment with a specific component's parameters early in the design process. Once you have determined the desired parameters, you can replace the "virtual"

component with a “real” component (for example, substitute a virtual diode with a 1N1199C diode).

For real components, the tab looks similar to this:



Note Refer to the *Virtual Components* section for information on virtual components.

Complete the following steps to edit the component in the database:

1. Click on **Edit Component in DB**.
2. Refer to the *Introduction to Component Editing* section of Chapter 6, *Component Editing*.

Resistors, Inductors and Capacitors

Complete the following steps to edit the value of a resistor, inductor, or capacitor:

1. Double-click on the component and select the **Value** tab.
2. Change the parameters as desired, and click **OK**.

Note For complete details on the contents of the **Value** tab for these types of components, refer to the *Component Reference help file*.

Edit Model Dialog Box

You can edit the model of a placed component using the **Edit Model** dialog box.

Note This procedure does not apply to resistors, inductors, or capacitors, as these are all generated using a common resistance, inductance, or capacitance model. Parameters for specific R, L, or C devices are set during or after part placement. Refer to the *Placing Resistors, Inductors or Capacitors* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Complete the following steps to edit the selected component's model:

1. Double-click the component to display its properties dialog box.
2. Click **Edit Model** in the **Value** tab. The **Edit Model** dialog box appears.
3. Edit the model in the listbox (unlabelled) as desired.
4. Make a selection based on the following:
   - **Current Instance Parameters** field—Available for standard Cadence® PSpice® models that contain paramaters (PARAMS). Enter edits to PARAMS here.
   - **Change Part Model**—Changes the model information for the selected component only. The button becomes active when a change is made to the model information.
   - **Change All Models**—Changes all models for the same part on the active workspace only, (that is, not in the database). The button becomes active when a change is made to the model information.
   - **Restore**—Restores the model to its original state. The button becomes active when a change is made to the model information.
   - **Cancel**—Closes the window without making the changes.
5. Click **OK** to save your settings and close the dialog box.

Caution Depending on your selection in the above step, changes made here apply to either the selected component, or all components for the same part on the active workspace. They do not apply to the same component in the database from which it was selected.

Edit Footprint Dialog Box

You can edit the footprint of a selected component using the **Edit Footprint** dialog box.

Complete the following steps to edit the selected component's footprint:

1. Double-click on the component and click on **Edit Footprint** in the component's **Value** tab. The **Edit Footprint** dialog box appears.
2. Click **Select From Database** to display the **Select a Footprint** dialog box.

   Select the desired footprint. Refer to the *Select a Footprint dialog box* section of Chapter 6, *Component Editing*, for information about using the **Select a Footprint** dialog box.

   Or

   Click **Change** to display the **Change Footprint** dialog box.

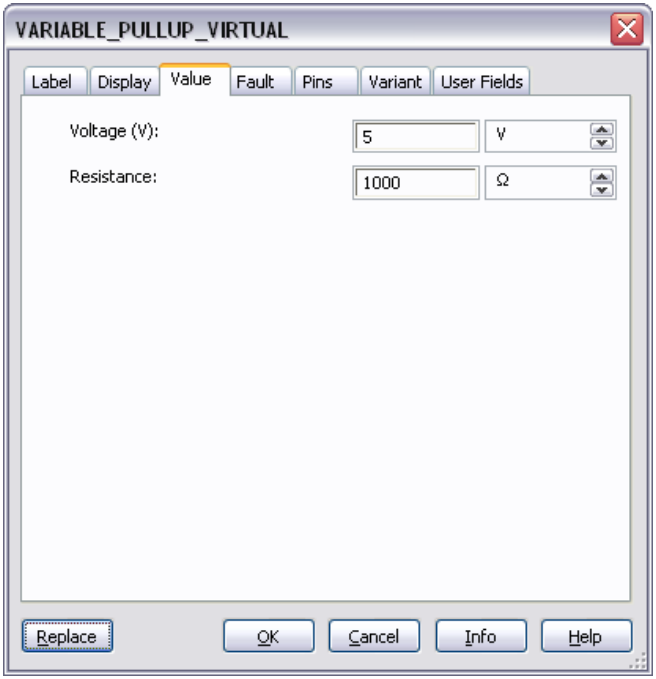
   Enter the desired **Footprint Manufacturer** and **Footprint Type** and click **OK** to return to the **Edit Footprint** dialog box.
3. Click **OK**.

Note Click **Map Pins** to display the **Advanced Pin Mapping** dialog. Refer to the *Advanced Pin Mapping Dialog* section of Chapter 6, *Component Editing*, for more information.

Virtual Components

Virtual components are not real; that is, you could not go to a supplier and purchase them. They have a symbol and a model, but no footprint. They are provided for your convenience to allow you to simulate "what-if" scenarios. Multisim treats them slightly differently from real components. By default, virtual components are shown in a different color from that of real components on your schematic. This is to remind you that, since they are not real, these components will not be exported to PCB layout software.

For virtual components, whose value can be set manually, the **Value** tab looks similar to this:



Note The contents of the **Value** tab change depending on the component selected.

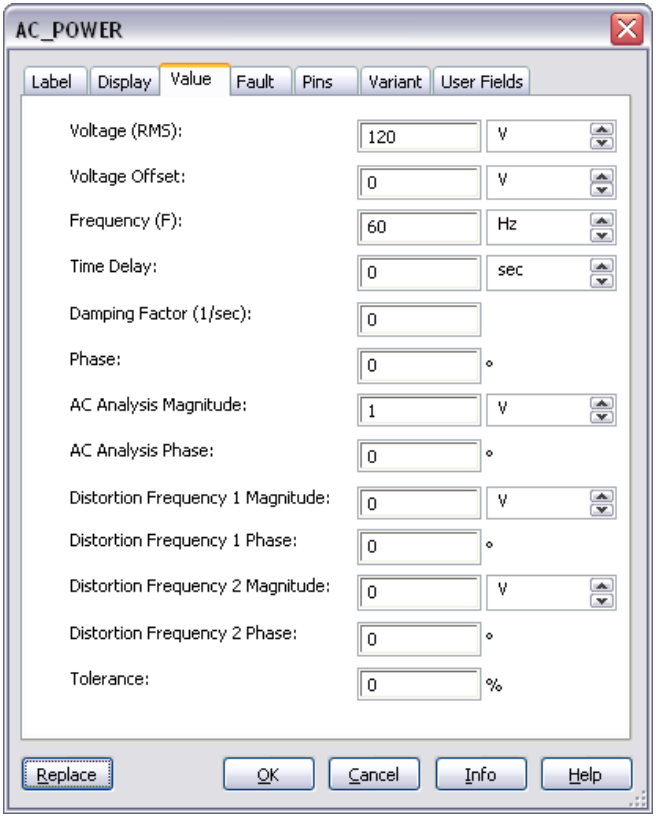
You can modify any of these fields.

Controlling How a Placed Component is Used in Analyses

For some components, such as power sources, you can determine how they are to be used in any analyses you might perform on the circuit. These components offer additional parameters in the **Value** tab.

Complete the following steps to control how the component is used in analyses:

1. Double-click on the component. The “properties” dialog box for the component appears.
2. Click the **Value** tab.



3. Modify the settings as desired.
4. Click **OK**.

Editing a Placed Component’s User Fields

Components have up to 20 user fields that provide user-specific information about the component (for example, vendor, manufacturer, hyperlink). The titles of the user fields are entered and edited from the **Database Manager**. Refer to the *Modifying User Field Titles and Content* section of Chapter 5, *Components*, for more information.

Complete the following steps to edit the contents of a placed component's user fields:

1. Double-click on the component to display its properties dialog box.
2. Click on the **User Fields** tab.
3. In the **Value** column, click in the field beside the desired user field **Title** and enter the desired information.

Assigning Faults to Components

You may want to assign faults to analog components for instructional purposes, such as troubleshooting exercises. You can manually assign faults to individual components in a circuit or let Multisim randomly assign faults to various components across a circuit.

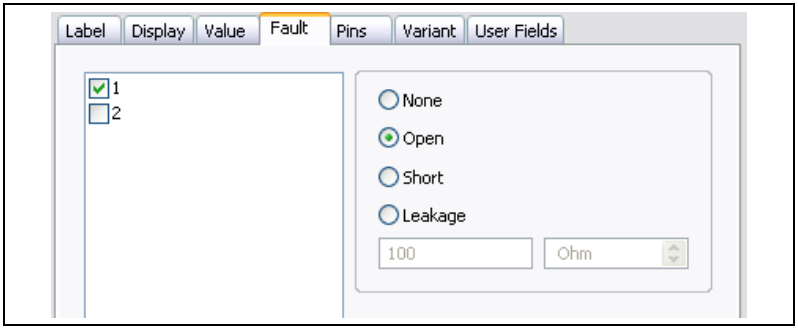
Note You cannot assign faults to digital components.

Setting a Placed Component's Faults

You can assign a fault to any terminal of the placed component using the **Fault** tab of that component's properties dialog box.

Complete the following steps to assign a fault to a placed component:

1. Double-click on the component. The component's properties dialog box appears.
2. Click the **Fault** tab and select the terminal(s) to which the fault should apply, as in the following example:



Note The contents of the **Fault** tab vary depending on the component type.

3. Enable the type of fault you want assigned to a terminal:
   - **None**—No fault.
   - **Open**—Assigns a very high resistance to the selected terminals, as if the wire leading to the terminals was broken.
   - **Short**—Assigns a very low resistance to the selected terminals, so the component has no measurable affect on the circuit.
   - **Leakage**—Assigns the resistance value specified in the fields below the option, in parallel with the selected terminals. This causes the current to leak past the terminals instead of going through them.
4. Click **OK** to save your changes.

Note You cannot assign faults to digital components.

Using the Auto Fault Option

When you use the **Auto Fault** option, you specify the number of any type of fault or, optionally, the number of faults per different type of fault, that you want Multisim to assign to placed components in the circuit.

Complete the following steps to use the auto fault option:

1. Choose **Simulate»Auto Fault Option**. The **Auto Fault** dialog box appears.

Note The **Auto Fault** option is disabled until a component is placed on the workspace.

2. Use the up/down keys or enter numerical values directly in the **Short**, **Open**, and **Leak** fields, or enter a numerical value in the **Any** field to let Multisim randomly select the type of faults to assign (in the quantity entered).
3. If you specify a number of leaks, enter a number and unit of measurement in the **Specify Leak Resistance** fields.
4. Click **OK** to apply the faults and return to the circuit window.

Note You cannot assign faults to digital components.

Spreadsheet View

The **Spreadsheet View** allows fast advanced viewing and editing of parameters including component details such as footprints, reference designators, attributes and design constraints and provides a global perspective on object properties.

Spreadsheet View Results Tab

If you select **Result Pane** in the **ERC Options** tab of the **Electrical Rules Check** dialog box, the **Results** tab will display the results of Electrical Rules Checks (ERCs). Refer to the *Electrical Rules Checking* and the *Result Pane* sections for more information.

The results of an **Edit»Find** command will also appear in the **Results** tab. Refer to the *Finding Components in Your Circuit* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Pop-up from the Results tab

Complete the following steps to use the pop-up menu on the results of an **ERC** or **Find**:

1. Right-click on the desired result to display the pop-up menu.
2. Select one of:
   - **Copy**—Copies entire contents of **Results** tab onto clipboard.
   - **Clear Results**—Clears content of **Results** tab.
   - **Go to**—Selects the item on the workspace.

Spreadsheet View Nets Tab

The **Nets** tab contains the following columns:

- **Net Name**—The net's name.
- **Sheet**—The filename of the sheet on which the net is found.
- **Color**—The net's color. **Default** is based on the color scheme selected in the **Circuit** tab of the **Sheet Properties** dialog box. Click to display a **Color** palette and select the desired color.
- **Trace Width**—Width of the traces after export to PCB layout. Unit of measure is set in Ultiboard.
- **Min Width**—Trace's minimum allowable width. Unit of measure is set in Ultiboard. Select desired row and then click in the field to edit.

- **Max Width**—Trace's maximum allowable width. Unit of measure is set in Ultiboard. Select desired row and then click in the field to edit.
- **Min Length**—Trace's minimum allowable length. Unit of measure is set in Ultiboard. Select desired row and then click in the field to edit.
- **Max Length**—Trace's maximum allowable length. Unit of measure is set in Ultiboard. Select desired row and then click in the field to edit.
- **Trace to Trace**—Minimum allowable space between traces in the net and traces in any other net on the PCB (printed circuit board). Unit of measure is set in Ultiboard. Click and type to make changes.
- **Trace to Pad**—Minimum allowable space between traces in the net and pads on any other net on the PCB. Unit of measure is set in Ultiboard. Click and type to make changes.
- **Trace to Via**—Minimum allowable space between traces in the net and vias on any other net on the PCB. Unit of measure is set in Ultiboard. Click and type to make changes.
- **Trace to Copper Area**—Minimum allowable space between traces in the net and copper areas on the PCB. Unit of measure is set in Ultiboard. Click and type to make changes.
- **Routing Layer**—The copper layer where the net will be placed. Click to display a drop-down list with the available selections. This drop-down is populated based on the selections made in the **PCB** tab of the **Sheet Properties** dialog box in the **Number of Copper Layers** field. Refer to the *Sheet Properties - PCB Tab* section of Chapter 1, *User Interface*, for more information.
- **Net Group**—Click in this field to enter a group for a net. This group can be used in Ultiboard to keep nets together during the PCB layout process.
- **IC Initial Condition**—The net's initial condition for **Transient Analysis** in Multisim.
- **NODESET**—The net's initial condition for **DC Operating Point** Analysis in Multisim. Helps the program find the DC or initial solution by making a preliminary pass with the specified nodes held to the given voltages. The restriction is then released and the iteration continues to the true solution. The **.NODESET** line may be necessary for convergence on bistable or a-stable sircuits. Refer to the *DC Operating Point Analysis* section of Chapter 1, *User Interface*, for more information.
- **Type**—Type of net. Can be **Power**, **Ground** or **Signal**.
- **Net Specific Setting**—If **Use Net-specific Setting** in the **Sheet Properties** dialog box is enabled, (refer to the *Sheet Properties -*

Circuit Tab section of Chapter 1, *User Interface*), the setting for the selected net is entered here. Choices are **Show Net Name** or **Hide Net Name**.

Spreadsheet View Components Tab

The **Components** tab contains the following columns:

- **RefDes**—The component's unique identifier (Reference Designator).
- **Sheet**—The filename of the sheet on which the component appears.
- **Section**—The section of a multi-section component such as a quad NAND gate.
- **Section Name**—The name of the section of a multi-section component.
- **Family**—The component's database family.
- **Value**—The component's value, for example, 5 V for a battery; or the component's model, for example, 2N2222A. Double-click to edit the component's model.
- **Tolerance**—The component's tolerance. Refer to the *Component Tolerances in Multisim* section of Chapter 1, *User Interface*, for more information.
- **Manufacturer**—The component's manufacturer; either **Generic** or a specific company.
- **Footprint**—The physical footprint of the component. Click on the field to change the footprint. Refer to the *Editing a Component's Footprint* section of Chapter 6, *Component Editing*, for more information.
- **Description**—The component's description.
- **Label**—The component's user-defined label. Click on the field and type in desired text. Can also be entered in the **Label** tab in the component's properties dialog box. Refer to the *Modifying Component Labels and Attributes* section of Chapter 2, *Schematic Capture - Basics*, for more information.
- **Coordinate X/Y**—The position of the component on the workspace. This field is read-only and changes as the component is moved on the workspace.
- **Rotation**—Click to display a drop-down list of the selections available to rotate the component. **Unrotated** is the component's original position. Other selections are: **Rotated 90** (90 degrees clockwise from original position), **Rotated 180** (180 degrees clockwise from original position), **Rotated –90** (90 degrees counter-clockwise from original

position). You can also rotate a component by right-clicking on it in the workspace. Refer to the *Pop-up Menus* section of Chapter 1, *User Interface*, for more information.

- **Flip**—Click to display a drop-down list of the selections available to flip the component. **Unflipped** is the component's original position. Other selections are: **Flipped X** (horizontal flip from original position), **Flipped Y** (vertical flip from original position), **Flipped XY** (a horizontal and a vertical flip from original position). You can also flip a component by right-clicking on it in the workspace. Refer to the *Pop-up Menus* section of Chapter 1, *User Interface*, for more information.
- **Color**—Component's color. **Default** is based on the color scheme selected in the **Circuit** tab of the **Sheet Properties** dialog box. Click to display a **Color** palette and select the desired color.
- **Component Spacing**—Minimum distance between the component and another component when using the shove option in Ultiboard. Unit of measure is set in the **PCB** tab of the **Sheet Properties** dialog box. Click to enter new data.
- **Component Group**—Click in this field to enter a group for a component. This group can be used in Ultiboard to keep components together during the PCB layout process.
- **Pin Swap**—If enabled, allows pins for like-components to be swapped during the PCB layout process. Click to toggle between **Yes** and **No**.
- **Gate Swap**—If enabled, allows gates with same functionality, such as two NAND gates to be swapped during PCB layout process. Click to toggle between **Yes** and **No**.
- **Fault**—Double-click to display the **Change Fault** dialog box. This is the same information that is available in the **Fault** tab in the component's properties dialog box.

Note You cannot assign faults to digital components.

- **VCC**—Supply voltage. If not blank, indicates that the part has the corresponding input. You may choose to assign a netname from those power and ground nets that are placed on the schematic.
- **VDD**—Supply voltage. If not blank, indicates that the part has the corresponding input. You may choose to assign a netname from those power and ground nets that are placed on the schematic.

- **VEE**—Supply voltage. If not blank, indicates that the part has the corresponding input. You may choose to assign a netname from those power and ground nets that are placed on the schematic.
- **VPP**—Supply voltage. If not blank, indicates that the part has the corresponding input. You may choose to assign a netname from those power and ground nets that are placed on the schematic.
- **GND**—Ground. If not blank, indicates that the part has the corresponding input. You may choose to assign a netname from those power and ground nets that are placed on the schematic.
- **VSS**—Usually ground, but for some components can be a negative supply voltage. If not blank, indicates that the part has the corresponding input. You may choose to assign a netname from those power and ground nets that are placed on the schematic.
- **Variant**—Displays a checkbox for each available variant. Enable the checkbox for each variant of the circuit that you wish the component to be included in. Refer to the *Variants* section of Chapter 4, *Working with Larger Designs*, for more information.

Spreadsheet View PCB Layers Tab

The **PCB Layers** tab contains the following columns:

- **Layer Name**—The contents of this column are set from the **PCB** tab of the **Sheet Properties** dialog box. Refer to the *Sheet Properties - PCB Tab* section of Chapter 1, *User Interface*, for more information.
- **Routable**—When enabled, the layer can be routed during the PCB layout process. Click to toggle between **Yes** and **No**.
- **Type**—Defines the type of layer. Double-click to display a drop-down list and choose from **Signal**, **Power**, **Ground** or **Unassigned**.

Spreadsheet View Simulation Tab

The **Simulation** tab is where errors and warnings from SPICE netlist checks, as well as simulation errors appear.

Refer to the *Netlist and Simulation Errors* section of Chapter 8, *Simulation*, for more information.

Spreadsheet View Buttons

The following buttons are available in the **Spreadsheet View**.

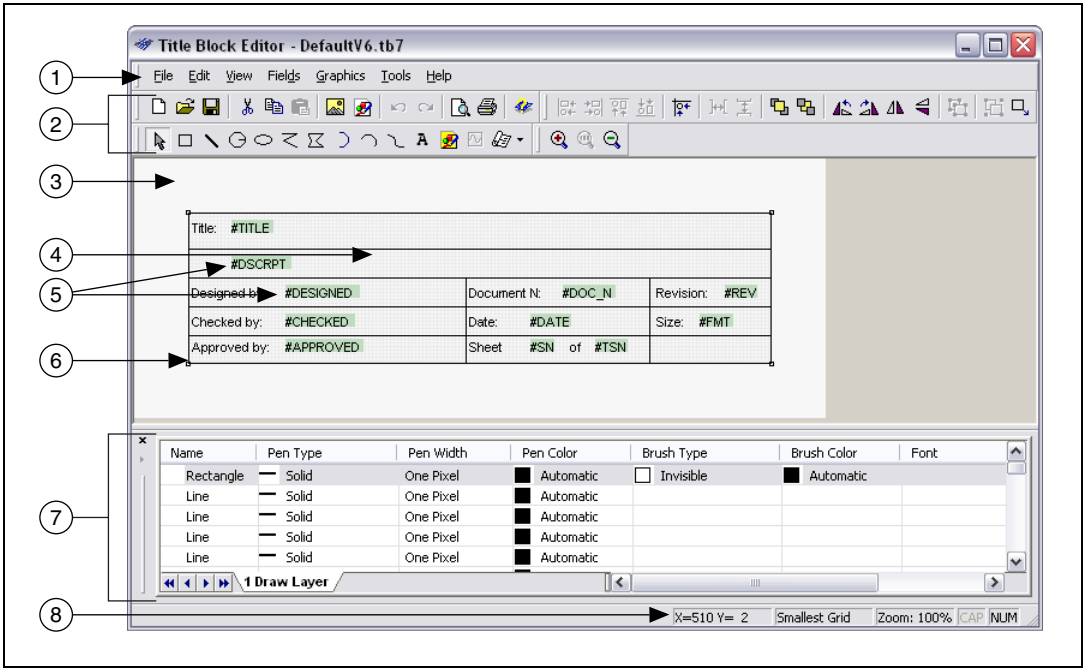
Button	Description
	Export to Textfile button. Displays a standard Windows Save dialog where you save the selection as a textfile.
	Export to CSV File button. Displays a standard Windows Save dialog where you save the selection as a file with comma-separated values.
	Export to Excel button. Click to open a Microsoft® Excel spreadsheet with the selected data displayed. (You must have Excel installed to use this function).
	Sort Ascending button. Sorts the selected column in ascending order.
	Sort Descending button. Sorts the selected column in descending order.
	Print button. Prints the data in the selected tab.
	Select All button. Selects all elements in the spreadsheet.
	Copy button. Copies the selection to the clipboard.
	Find and Select button. Finds and highlights the selected component or net on the workspace. This feature is not available in all versions of Multisim.

Button	Description
	All button. Displays all of the netlists or components (depending on the selected tab) from all sheets, multi-pages, subcircuits and hierarchical blocks in the current design.
	Replace Selected Components button. Before using, select the desired component(s) in the circuit window to be replaced. Invokes the Select a Component browser from which you can select a new component. Click **OK** to replace the old component(s) with the selected new one.

Title Block Editor

The **Title Block Editor** is a specialized graphics editor that allows you to create or modify a title block. For example, you can insert and position title block data, change font properties and place or move graphic objects.

The **Title Block Editor** looks like this:



1	Menu Bar	3	Workspace	5	Fields	7	Spreadsheet View
2	Toolbars	4	Draw Grid	6	Boundary Box	8	Status Bar

The **Title Block Editor** consists of:

- the **Menu Bar** (1), which contains the menus with their associated commands.
- the **Toolbars** (2), which give quick access to the most commonly-used tools.
- the **Workspace** (3), which is where you build or modify your title blocks. The **Draw Grid** (4) aids in the placement of graphic elements and **Fields** (5) inside the **Boundary Box** (6).
- the **Spreadsheet View** (7), which is where you find and edit various title block parameters.
- the **Status Bar** (8), which gives information on the currently selected object or action.

Complete the following steps to edit a title block that is already in your circuit:

1. Right-click on the desired title block and select **Edit Title Block** from the pop-up. The **Title Block Editor** appears with the selected title block loaded.

Note **In-Place Edit Mode** displays at the bottom of the dialog box when the **Title Block Editor** is launched using the method described above. Changes will apply to the selected title block only.

2. Edit the title block as described in:
   - *Enter Text Dialog Box*.
   - *Placing Fields*.
   - *Title Block Editor Spreadsheet View*.
   - *Title Block Editor Menus*.
   - *Title Block Editor Toolbars*.
3. Select **File»Exit** and click **Yes** when prompted to save changes. The **Title Block Editor** closes and you are returned to the main Multisim workspace. The changes are reflected in the title block.

Complete the following steps to edit a title block that is stored in the **Title Block** folder, or to create a new title block:

1. Select **Tools»Title Block Editor**. The **Title Block Editor** appears with a new un-named title block.
2. To create a new title block, start working from here.

   Or

   To edit an existing title block, select **File»Open**, navigate to the **Titleblocks** folder, select the desired title block and click **Open**.
3. Edit the title block using the menus and toolbars as described in the following sections.
4. Select **File»Exit** and click **Yes** when prompted to save your changes. If this is a new title block, a standard **Save As** dialog box appears. Enter the desired filepath and filename, and click **Save**.

   If it is an existing title block the changes are saved and the **Title Block Editor** closes.

The following sections describe the **Title Block Editor** functionality in more detail.

Enter Text Dialog Box

The **Enter Text** dialog box is used to enter and format text and place it on the title block.

Complete the following steps to enter text on a title block:

1. Select **Graphics»Text** to display the **Enter Text** dialog box.
2. Type the desired text in the **Enter Text** field.
3. Change the formatting of the text as desired:
   - **Font** field—Select desired font from the list or type in.
   - **Font Style** field—Select desired style from the list.
   - **Size** field—Select desired size from the list or type in.
   - **Text Orientation** box—Select either horizontal or vertical orientation.
   - **Automatic** drop-down list—Optionally, select a new color from the pop-up that appears when you click on the down-arrow.
4. Click **OK**, move the cursor to the desired location and click the mouse to place the text.

Placing Fields

Fields are placeholders for text that appears in the title block on your circuit schematic.

Note The actual text that appears in these fields is entered in the **Title Block** dialog box, which is accessed from Multisim's main screen.

For a field's text to appear in the title block in Multisim, you must first place the field for that text in the title block using the **Title Block Editor**.

Complete the following steps to place a field on the title block:

1. Select the desired field type from the **Fields** menu (for example, **Revision**).

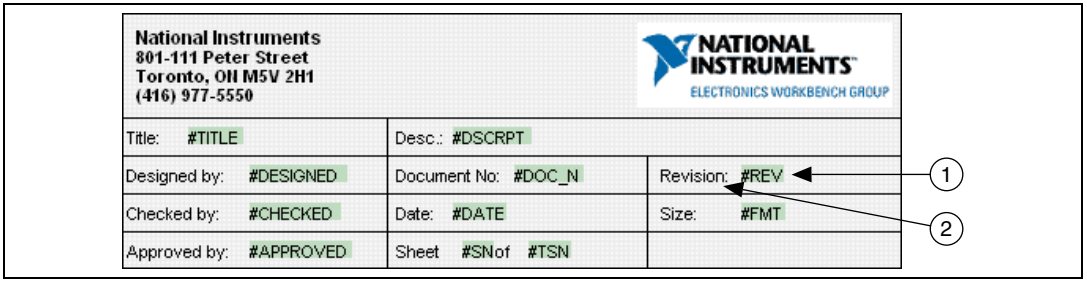
   Or

   Click on the **Text Field** button in the **Draw Tools** toolbar and select the desired field type from the pop-up that displays (for example, **Revision**).

The **Enter Title Block Attribute** dialog box displays the code for the selected field in the **Title Block Attribute** field. (Since we selected **Revision** in this example, `#REV` appears.)

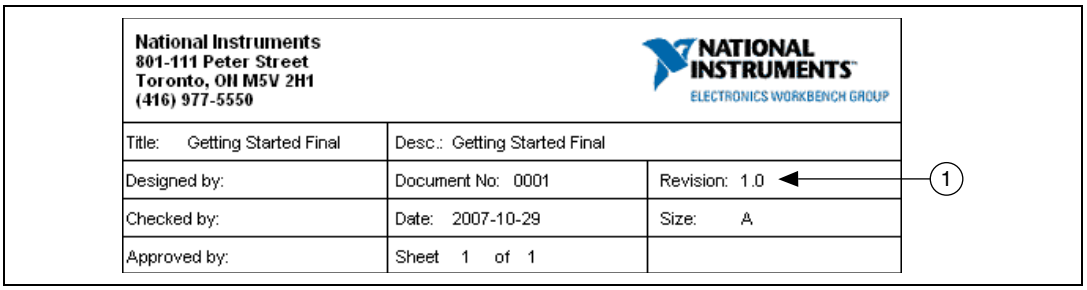
Note Refer to the *Field Codes* section for a list of all field codes.

2. Change the formatting of the text as desired:
   - **Font** field—Select desired font from the list or type in.
   - **Font Style** field—Select desired style from the list.
   - **Size** field—Select desired size from the list or type in.
   - **Text Orientation** box—Select either horizontal or vertical orientation.
   - **Automatic** drop-down—Optionally, select a new color from the pop-up that appears when you click on the down-arrow.
3. Click **OK**, move the cursor to the desired location and click the mouse to place the field. All placed *fields* appear in green highlighting, shown in the example below (1). Placed *text* (2) is not highlighted.



1 Code for Placed Revision *field* 2 Placed *text* is not Highlighted

4. After completing any other edits to the title block, select **File»Exit** and save the changes when prompted. You are returned to the main Multisim screen, where the field code (`#REV`) has been replaced by text, as shown in the example below (1). This text is set in the **Revision** field of the **Title Block** dialog box, which is found in the main Multisim application. Refer to the *Entering the Title Block Contents* section of Chapter 2, *Schematic Capture - Basics*, for more information.



1 Field Code Replaced by Text

Caution The width of a *field* as displayed in the **Title Block Editor** is not the same as the actual *text* that is placed in that field using the **Title Block** dialog box (refer to the *Entering the Title Block Contents* section of Chapter 2, *Schematic Capture - Basics*, for more information). Space used is also dependant on the font size. If after placing the actual text in the title block using the **Title Block** dialog box, you find that text overlaps, you must either adjust the text, or return to the **Title Block Editor** and adjust the positioning of the fields.

Note If text in the **Title Block** dialog box does not appear in your title block, it is because the field corresponding to that text was not placed in the title block using the **Title Block Editor**.

Field Codes

The following are the available field codes:

- **Title**—`#TITLE`
- **Description**—`#DSCRPT`
- **Designed By**—`#DESIGNED`
- **Checked By**—`#CHECKED`
- **Approved By**—`#APPROVED`
- **Document Number**—`#DOC_N`
- **Date**—`#DATE`
- **Current Sheet Number**—`#SN`
- **Total Sheet Numbers**—`#TSN`
- **Revision**—`#REV`
- **Format**—`#FMT`
- **Custom Field 1**—`#CUSTOM_1`
- **Custom Field 2**—`#CUSTOM_2`

- **Custom Field 3**—`#CUSTOM_3`
- **Custom Field 4**—`#CUSTOM_4`
- **Custom Field 5**—`#CUSTOM_5`

Title Block Editor Spreadsheet View

The **Spreadsheet View** is where you find and edit various title block parameters. When you select an item on the workspace, it is highlighted in the spreadsheet, and vice versa. If you make a change to an item in the spreadsheet, it is reflected on the title block in the workspace.

The following columns are found in the **Title Block Editor Spreadsheet View**:

- **Name**—The type of graphic element.
- **Pen Type**—The appearance of lines for graphics elements, excluding placed text. Select the desired row and click to display a list of pen types. Choices are: **Solid**; **Dash**; **Dot**; **Dash-Dot**; **Dash-Dot-Dot**; **Invisible**; **Solid Inside Frame**.
- **Pen Width**—The width of the lines in graphic elements, excluding placed text. Select the desired row and click to display a list of pen types. Choices are: **Not Scaleable**; **One Pixel**; **Two Pixels**; **Three Pixels**; **Four Pixels**; **Five Pixels**.
- **Pen Color**—The color of lines for graphic elements, including placed text. Select the desired row and click to display a color palette.
- **Brush Type**—The style of the fill in elements such as polygons, that have a fill. Select the desired row and click to display a list of fill types. Choices are: **Solid**; **Invisible**; **Horizontal**; **Vertical**; **Diagonal Downward**; **Diagonal Upward**; **Cross**; **Diagonal Cross**.
- **Brush Color**—The color of the fill in elements such as polygons, that have a fill. Select the desired row and click to display a color palette.
- **Font**—The font name, active for placed text elements only. Double-click to display a list of fonts.
- **Font Style**—The font style, active for placed text elements only. Select the desired row and click to display a list. Choices are: **Regular**; **Italic**; **Bold**; **Bold Italic**.
- **Font Size**—The font size, active for placed text elements only. Select the desired row and click to display a list of sizes.

Note If objects are grouped using **Edit»Group**, their distinct names and properties no longer appear in the **Name** column. The name for any grouped object appears as **Group**.

Title Block Editor Menus

The **Title Block Editor** menus contain the commands necessary to create and edit title blocks.

File Menu

The following selections are available under the **File** menu:

- **New**—Opens a new untitled document in the **Title Block Editor**. If you already have one open, it will close first, after prompting you to save any changes.
- **Open**—Opens an existing document in the **Title Block Editor**. If you already have one open, it will close first, after prompting you to save any changes.
- **Save**—Saves changes to the active document.
- **Save As**—Opens the standard Windows "Save As" dialog box where you can save the active document under a new or existing name.
- **Print Setup**—Opens the standard Windows "Print Setup" dialog box where you can enter the desired parameters for your printer.
- **Print Preview**—Opens the **Print Preview** dialog box, which shows the title block that is in the active document.
- **Print**—Opens the standard Windows "Print" dialog box where you can enter the desired printing properties and print the title block. The title block is printed with the best fit to the page. There are no other magnifications available for printing the title block.
- **Exit**—Closes the **Title Block Editor** and returns you to the main Multisim screen. Before exiting, you are prompted to save any changes to the active document.

Edit Menu

The following selections are available under the **Edit** menu:

- **Undo**—Undoes the previous action.
- **Redo**—Undoes the previous "undo" action.
- **Cut**—Removes the selected element(s) from the workspace and places them on the clipboard.
- **Copy**—Places a copy of the selected element(s) on the clipboard.
- **Paste**—Places a copy of the element(s) on the clipboard on the workspace at the cursor's location.

- **Delete**—Removes the selected element(s) from the workspace. They are *not* placed on the clipboard.
- **Copy As Picture**—Copies the title block on the workspace as a metafile.
- **Copy As Bitmap**—Copies the title block on the workspace as a bitmap image.
- **Select All**—Selects all of the elements on the workspace.
- **Flip Horizontal**—Flips the selected element(s) horizontally.
- **Flip Vertical**—Flips the selected element(s) vertically.
- **Rotate 90 Clockwise**—Rotates the selected element(s) 90 degrees clockwise.
- **Rotate 90 Counter CW**—Rotates the selected element(s) 90 degrees counter-clockwise.
- **Snap To Grid**—Snaps the selected element(s) to the Draw Grid that is found within the title block's boundary box.
- **Group**—Places selected elements in one group.
- **Ungroup**—Returns an element that was made using the **Group** command, back to its individual elements.
- **Bring To Front**—Brings selected element(s) to the foreground on the workspace. Other element(s) appear behind them.
- **Send To Back**—Sends selected element(s) to the background on the workspace. Other elements appear in front of them.
- **Resize Boundary Box**—Places a cursor at the lower-right side of the boundary box. Drag it to the desired location to resize the boundary box. You cannot make the boundary box smaller than the elements that it contains.

View Menu

The following selections are available under the **View** menu:

- **Toolbars**—Toggles the following toolbars on and off: **Standard Toolbar**; **Zoom Toolbar**; **Draw Tools**; **Drawing Toolbar**.
- **Spreadsheet**—Toggles the Title Block Editor Spreadsheet View on and off. Refer to the *Title Block Editor Spreadsheet View* section for more information.
- **Status Bar**—Toggles the **Status Bar** at the bottom of the screen on and off.

- **Show Draw Grid**—Toggles the **Draw Grid**, which displays inside the **Boundary Box**, on and off.
- **Draw Grid Size**—Sets the size of the **Draw Grid**, which displays inside the **Boundary Box**. The choices are: **No Grid** (select if you wish to draw an element that does not snap to the grid); **Smallest Grid**; **Small Grid**; **Regular Grid**; **Large Grid**.
- **Zoom In**—Magnifies the elements(s) in the workspace.
- **Zoom Out**—Reduces the viewing size of the elements(s) in the workspace.
- **Zoom 100%**—Displays the items in the workspace at their normal viewing size. This is the size that they will be displayed at in Multisim. When the **Title Block Editor** first opens, the magnification is set to 100%.
- **Center By Mouse**—When viewing the workspace at high magnifications, you can use this command to center the image on the workspace. Select **Center By Mouse** and then click on the spot that you would like to be placed at the center of the workspace.
- **Redraw**—Redraws all elements in the workspace.

Fields Menu

Fields are placeholders for text that appear in the title block on Multisim's main screen. Refer to the *Placing Fields* section for more information.

The following selections are available under the **Fields** menu:

- **Select**—Lets you select specific element(s) on the workspace.
- **Title**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Title** field (`#TITLE`) and click **OK** to place the field on the title block.
- **Description**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Description** field (`#DSCRPT`) and click **OK** to place the field on the title block.
- **Designed By**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Designed By** field (`#DESIGNED`) and click **OK** to place the field on the title block.
- **Checked By**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Checked By** field (`#CHECKED`) and click **OK** to place the field on the title block.

- **Approved By**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Approved By** field (#APPROVED) and click **OK** to place the field on the title block.
- **Document Number**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Document Number** field (#DOC_N) and click **OK** to place the field on the title block.
- **Date**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Date** field (#DATE) and click **OK** to place the field on the title block.
- **Current Sheet Number**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Current Sheet Number** field (#SN) and click **OK** to place the field on the title block.
- **Total Sheet Numbers**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Total Sheet Numbers** field (#TSN) and click **OK** to place the field on the title block.
- **Revision**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Revision** field (#REV) and click **OK** to place the field on the title block.
- **Format**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for the **Format** field (#FMT) and click **OK** to place the field on the title block.
- **Custom Field 1**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for **Custom Field 1** (#CUSTOM_1) and click **OK** to place the field on the title block.
- **Custom Field 2**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for **Custom Field 2** (#CUSTOM_2) and click **OK** to place the field on the title block.
- **Custom Field 3**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for **Custom Field 3** (#CUSTOM_3) and click **OK** to place the field on the title block.
- **Custom Field 4**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for **Custom Field 4** (#CUSTOM_4) and click **OK** to place the field on the title block.
- **Custom Field 5**—Displays the **Enter Title Block Attribute** dialog box, where you enter formatting information for **Custom Field 5** (#CUSTOM_5) and click **OK** to place the field on the title block.

Graphics Menu

The following selections are available under the **Graphics** menu:

- **Text**—Displays the **Enter Text** dialog box where you enter and format text to be placed on the workspace. Refer to the *Enter Text Dialog Box* section for more information.
- **Line**—Draws a line on the workspace.
- **Multiline**—Draws a multiline on the workspace.
- **Half Ellipse Arc**—Places half of an ellipse on the workspace. Click once at the desired starting point and click again where you wish the diameter of the ellipse to end. As you move the cursor, a dotted outline of the ellipse appears. Click again to place the outer point of the arc at the desired location.
- **Segment Arc**—Places an arc on the workspace. Click once to place the center point of the arc, click again to place the outer diameter point. As you move the cursor, a dotted outline of the arc appears. Move the cursor and click to place the arc's end point.
- **Bezier**—Places a bezier curve on the workspace. Click to place the start of the curve, then move the cursor on the workspace. As you move the cursor, a dotted outline of the bezier curve appears. Click to place the end point of the bezier, then move and click the cursor twice more to form the final shape of the bezier.
- **Rectangle**—Places a rectangle on the workspace.
- **Circle**—Places a circle on the workspace. Click to place the center of the circle, then move the cursor on the workspace. As you move the cursor, a dotted outline of the circle appears. When the circle is the desired shape and size, click to place it on the workspace.
- **Ellipse**—Places an ellipse on the workspace. Click to place the center of the ellipse, then move the cursor on the workspace. As you move the cursor, a dotted outline of the ellipse appears. When the ellipse is the desired shape and size, click to place it on the workspace.
- **Polygon**—Places a polygon on the workspace.
- **Bitmap**—Places a bitmap image on the workspace. Click to display a standard Windows "Open" dialog box, where you can select the desired bitmap.

Tools Menu

The following selection is available under the **Tools** menu:

- **Customize**—Displays the **Customize** dialog box. Refer to the *Customizing the Interface* section of Chapter 1, *User Interface*, for more information.

Help Menu

The following selections are available under the **Help** menu:

- **Help Topics**—Displays the help file.
- **About Title Block Editor**—Displays a splash dialog with information about the Title Block Editor.

Pop-up Menus

Depending on where you right-click in the **Title Block Editor**, different context-sensitive pop-up menus appear.

Right-click on the menu/toolbar area to display a pop-up which allows you to toggle the following toolbars on and off: **Standard Toolbar**; **Zoom Toolbar**; **Draw Tools**; **Drawing Toolbar**.

Right-click in the *workspace* to display a pop-up that contains: **Cut**; **Copy**; **Paste**; **Show Draw Grid**; **Draw Grid Size**; **Snap To Grid**; **Flip Horizontal**; **Flip Vertical**; **Rotate 90 Clockwise**; **Rotate 90 Counter CW**. Refer to the *Edit Menu* section for more information. If you do not right-click on a specific item in the workspace, items in the pop-up will be greyed-out.

Title Block Editor Toolbars

The toolbars give access to the most commonly-used tools in the **Title Block Editor**.

Standard Toolbar - Title Block Editor

The buttons in the **Standard** toolbar found in the **Title Block Editor** are described below:

Button	Description
	New button. Opens a new untitled document in the **Title Block Editor**. If you already have one open, it will close first, after prompting you to save any changes.
	Open button. Opens an existing document in the **Title Block Editor**. If you already have one open, it will close first, after prompting you to save any changes.
	Save button. Saves changes to the active document.
	Cut button. Removes the selected element(s) from the workspace and places them on the clipboard.
	Copy button. Places a copy of the selected element(s) on the clipboard.
	Paste button. Places a copy of the element(s) on the clipboard on the workspace at the cursor's location.
	Copy As Picture button. Copies the title block on the workspace as a metafile.
	Copy As Bitmap button. Copies the title block on the workspace as a bitmap image.
	Undo button. Undoes the previous action.
	Redo button. Redoes the previous "undo" action.
	Print Preview button. Opens the **Print Preview** dialog box, which shows the title block in the active document with the best fit to the page. There are no other sizes available.

Button	Description
	Print button. Opens the standard Windows “Print” dialog box, where you can enter the desired printing properties and print the title block. The title block is printed with the best fit to the page. There are no other magnifications available for printing the title block.
	About Title Block Editor button. Displays an “About” box with information about the Title Block Editor.

Zoom Toolbar - Title Block Editor

The buttons in the **Zoom** toolbar found in the **Title Block Editor** are described below:

Button	Description
	Zoom In button. Magnifies the elements on the workspace.
	Zoom 100% button. Displays the items in the workspace at their normal viewing size. This is the size that they will be displayed at in Multisim. When the **Title Block Editor** first opens, the magnification is set to 100%.
	Zoom Out button. Reduces the viewing size of the element(s) on the workspace.

Draw Tools Toolbar - Title Block Editor

The buttons in the **Draw Tools** toolbar found in the **Title Block Editor** are described below:

Button	Description
	Select button. Use to select element(s) on the workspace by clicking and dragging the mouse.
	Rectangle button. Places a rectangle on the workspace.

Button	Description
	Line button. Places a line on the workspace.
	Circle button. Places a circle on the workspace. Click to place the center of the circle, then move the cursor on the workspace. As you move the cursor, a dotted outline of the circle appears. When the circle is the desired shape and size, click to place it on the workspace.
	Ellipse button. Places an ellipse on the workspace. Click to place the center of the ellipse, then move the cursor on the workspace. As you move the cursor, a dotted outline of the ellipse appears. When the ellipse is the desired shape and size, click to place it on the workspace.
	Multiline button. Places a multiline on the workspace.
	Polygon button. Places a polygon on the workspace.
	Half Ellipse Arc button. Places a half ellipse arc on the workspace.
	Segment Arc button. Places an arc on the workspace. Click once to place the center point of the arc, click again to place the out diameter point. As you move the cursor, a dotted outline of the arc appears. Move the cursor and click to place the arc's end point.
	Bezier button. Places a bezier curve on the workspace.
	Text button. Displays the **Enter Text** dialog box, where you enter and format text to be placed on the workspace. Refer to the *Enter Text Dialog Box* section for more information.

Button	Description
	Bitmap button. Places a bitmap image on the workspace. Click to display a standard Windows “Open” dialog box, where you can select the desired bitmap.
	Text Field button. Select to enter formatting information for a specific field. Displays a drop-down list with these choices: Title; Description; Designed By; Checked By; Approved By; Document Number; Date; Current Sheet Number; Total Sheet Numbers; Revision; Format; Custom Field 1; Custom Field 2; Custom Field 3; Custom Field 4; Custom Field 5. When a choice is made, the Enter Title Block Attribute dialog box displays. Refer to the *Placing Fields* section for more information.

Drawing Toolbar - Title Block Editor

The buttons on the **Drawing** toolbar in the **Title Block Editor** are described below:

Button	Description
	Align Left button. Moves selected objects horizontally so that their left sides line up with the left side of the left-most object. At least two objects must be selected to enable this button.
	Align Right button. Moves selected objects horizontally so that their right sides line up with the right side of the right-most object. At least two objects must be selected to enable this button.
	Align Top button. Moves selected objects vertically so that their top sides line up with the top side of the top-most object. At least two objects must be selected to enable this button.
	Align Bottom button. Moves the selected objects vertically so that their bottom sides line up with the bottom side of the bottom-most object. At least two objects must be selected to enable this button.
	Snap To Grid button. Snaps the selected element(s) to the **Draw Grid** that is found within the title block’s boundary box.

Button	Description
	Distribute Horizontal button. Evenly spaces the selected objects horizontally. At least three objects must be selected to enable this button.
	Distribute Vertical button. Evenly spaces the selected objects vertically. At least three objects must be selected to enable this button.
	Bring To Front button. Brings selected element(s) to the foreground on the workspace. Other element(s) appear behind them.
	Send To Back button. Sends selected element(s) to the background on the workspace. Other elements appear in front of them.
	Rotate 90 Counter CW button. Rotates the selected element(s) 90 degrees counter-clockwise.
	Rotate 90 Clockwise button. Rotates the selected element(s) 90 degrees clockwise.
	Flip Horizontal button. Flips the selected element(s) horizontally on the workspace.
	Flip Vertical button. Flips the selected element(s) vertically on the workspace.
	Ungroup button. Returns an element that was made using the Group command back to its individual elements.
	Group button. Places selected elements in one group.
	Resize Boundary Box button. Places a cursor at the lower-right side of the boundary box. Drag it to the desired location to resize the boundary box. You cannot make the boundary box smaller than the elements that it contains or smaller than is required for the pins that are attached to it.

Electrical Rules Checking

Once you have wired your circuit, you can check the connections for correctness based on the rules set up in the **Electrical Rules Check** dialog box. **Electrical Rules Checking** creates and displays a report detailing connection errors (such as an output pin connected to a power pin) and unconnected pins.

Depending on your circuit, you may wish to have warnings issued if some types of connections are present, error messages for other connection types, and no warnings or errors for other connections. You control the type of connections that are reported when **Electrical Rules Checking** is done by setting up the rules in the grid found in the **ERC Rules** tab of the **Electrical Rules Check** dialog box.

When you run an electrical rules check (ERC), any anomalies are reported into a results pane at the bottom of the screen and the circuit is annotated with circular error markers. Double-clicking on an error in the results pane will center and zoom on the error location.

Complete the following steps to run the electrical rules check:

1. Select **Tools»Electrical Rules Check** to display the **Electrical Rules Check** dialog box.
2. Set up the reporting options as described in the *ERC Options Tab* and *ERC Rules Tab* sections.
3. Set up the rules as described in the *ERC Rules Tab* section.
4. Click **OK**. The results display in the format selected in the **Output** box in the **ERC Options** tab.

   In the following examples, a power pin has been connected to an output pin, which was defined as an error in the **ERC Rules** tab. All others pins have been left unconnected.

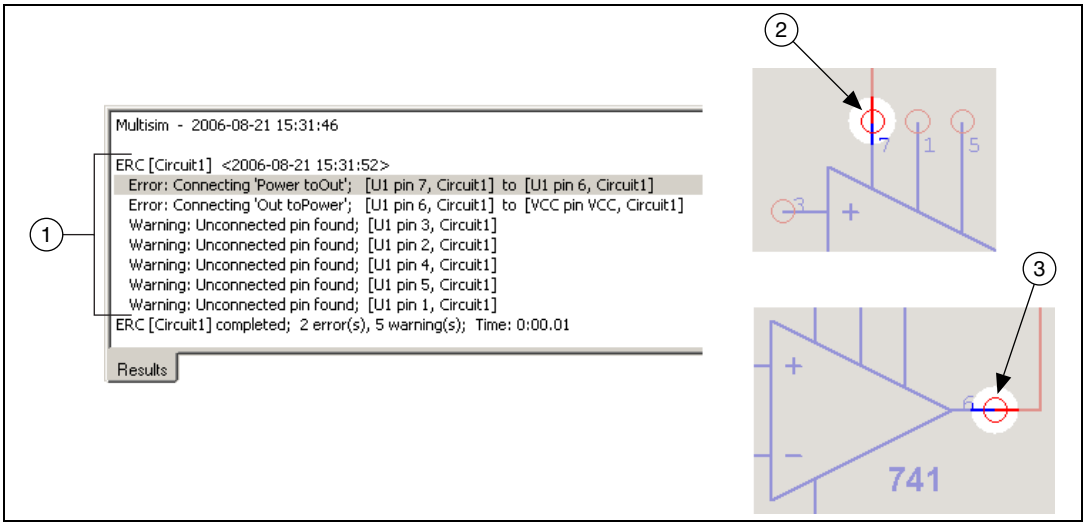
Note You can select whether or not to include specific pins in a component in the ERC. Refer to the *Component's Pins Tab* section for more information.

Result Pane

If you select **Result Pane** to display your output, errors and warnings are detailed in the **Results** tab of the **Spreadsheet View** as shown in the example in the figure below.

Double-click on individual errors or warnings in the ERC list (1). The error or warning is selected and zoomed and shown in (2) below. Double-click on the same error or warning to zoom to the other pin associated with the error or warning (3). This does not apply to unconnected pins.

The red circles are ERC markers, and indicate either errors or warnings.


1
2
3
Multisim - 2006-08-21 15:31:46
ERC [Circuit1] <2006-08-21 15:31:52>
Error: Connecting 'Power toOut'; [U1 pin 7, Circuit1] to [U1 pin 6, Circuit1]
Error: Connecting 'Out toPower'; [U1 pin 6, Circuit1] to [VCC pin VCC, Circuit1]
Warning: Unconnected pin found; [U1 pin 3, Circuit1]
Warning: Unconnected pin found; [U1 pin 2, Circuit1]
Warning: Unconnected pin found; [U1 pin 4, Circuit1]
Warning: Unconnected pin found; [U1 pin 5, Circuit1]
Warning: Unconnected pin found; [U1 pin 1, Circuit1]
ERC [Circuit1] completed; 2 error(s), 5 warning(s); Time: 0:00.01
Results
7
1
5
3
6
741

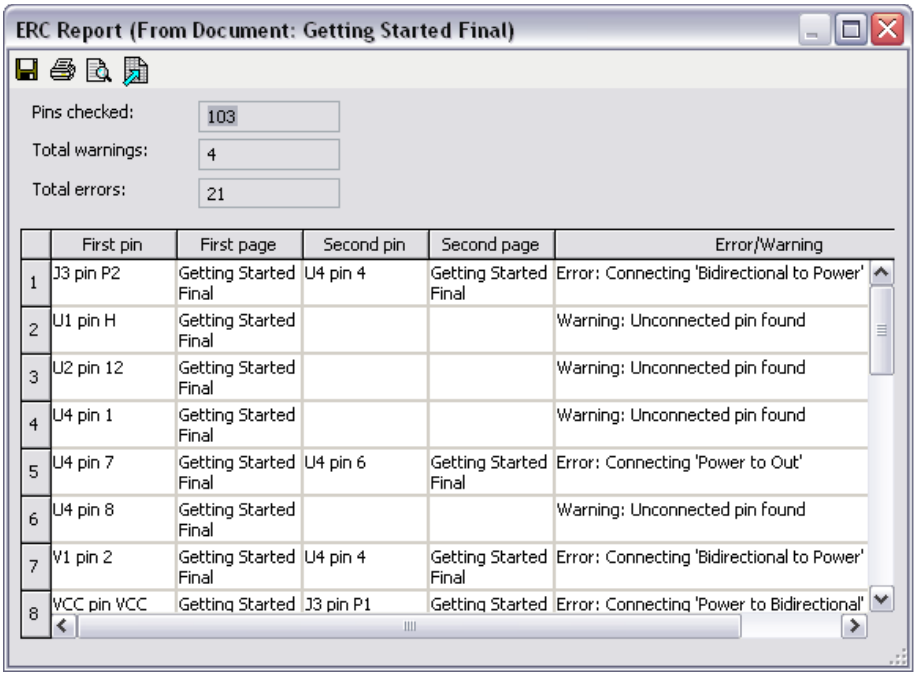
1 ERC List	2 Selected Error's First Pin	3 Selected Error's Second Pin

File

If you select **File** in the **Output** box, the results of the ERC are saved to a text file in the filepath and name that you enter in the **File** field.

List View

If you select **List View**, a report displays as shown in the example below:



	First pin	First page	Second pin	Second page	Error/Warning
1	J3 pin P2	Getting Started Final	U4 pin 4	Getting Started Final	Error: Connecting 'Bidirectional to Power'
2	U1 pin H	Getting Started Final			Warning: Unconnected pin found
3	U2 pin 12	Getting Started Final			Warning: Unconnected pin found
4	U4 pin 1	Getting Started Final			Warning: Unconnected pin found
5	U4 pin 7	Getting Started Final	U4 pin 6	Getting Started Final	Error: Connecting 'Power to Out'
6	U4 pin 8	Getting Started Final			Warning: Unconnected pin found
7	V1 pin 2	Getting Started Final	U4 pin 4	Getting Started Final	Error: Connecting 'Bidirectional to Power'
8	VCC pin VCC	Getting Started	J3 pin P1	Getting Started	Error: Connecting 'Power to Bidirectional'

Use the buttons detailed below as required:

Button	Description
	Save to a Text File button. Click to save the data in the dialog box to a text file. A standard Windows save dialog box appears. Choose the desired filepath and click **Save**.
	Print button. Click to display a standard Windows print dialog box. Choose the desired print options and click **OK**.
	Print Preview button. Click to display the **Print Preview** dialog box.
	Export to MS Excel button. Click to open a Microsoft® Excel spreadsheet with the data from the dialog box displayed. (You must have Excel installed to use this function.)

ERC Options Tab

This section describes how to set up **Electrical Rule Check** (ERC) options. Refer to the *Electrical Rules Checking* section for information about how to run an ERC.

Complete the following steps to set up the ERC options:

1. Select **Tools»Electrical Rules Check** to display the **Electrical Rules Check** dialog box and click on the **ERC Options** tab.
2. In the **Scope** box, select one of:
   - **Current Page**—Runs the ERC on the page displayed and selected on your workspace.
   - **Whole Design**—Runs the ERC on all subcircuits, hierarchical blocks and multi-pages associated with the current design.
3. In the **Flow Through** box, select as many of the following as desired:
   - **Off-Page Connectors**—Checks connections between pins connected through off-page connectors. Does not check any other pins on the associated multi-page unless **Check Touched Pages** is also selected.
   - **HB/SC Pins**—Checks connections between pins connected through HB/SC (hierarchical block or subcircuit) connectors. Does not check any other pins on the associated hierarchical block or subcircuit unless **Check Touched Pages** is also selected.
   - **Bus Off-Page Connectors**—Checks connections between pins connected to buses through bus off-page connectors. Does not check any other pins on the associated multi-page unless **Check Touched Pages** is also selected.
   - **Bus HB/SC Pins**—Checks connections between pins connected to buses through Bus HB/SC (hierarchical block or subcircuit) connectors. Does not check any other pins on the associated hierarchical block of subcircuit unless **Check Touched Pages** is also selected.
   - **Check Touched Pages**—Is active when one or more of the above are selected. When selected, ERC checks all connections on the associated multi-page, hierarchical block or subcircuit.

4. In the **Report Also** box, select the following as desired:
   - **Unconnected Pins**—Checks for pins that are not connected to anything.
   - **Excluded Pins**—Checks pins that have been excluded from ERC in the **Pins** tab of the component's properties dialog box. Refer to the *Component's Pins Tab* section for more information.
5. In the **ERC Marker** box, select the following as desired:
   - **Clear ERC Markers**—Clears existing ERC markers (red circles indicating errors and warnings) from the workspace when you run the ERC. Refer to the *Clearing ERC Markers* section for information about clearing ERC markers *without* running an ERC.
   - **Create ERC Markers**—Places red circles indicating errors and warnings on the workspace.
6. In the **Output** box, select one of:
   - **Result Pane**—Displays ERC results in the **Results** tab of the **Spreadsheet View**. If you select **Clear Pane**, any previous ERC results will be cleared from the **Results** tab when a new ERC is run. Refer to the *Result Pane* section for an example.
   - **File**—The results are saved in the filepath and name that you enter in the **File** field. Refer to the *File* section for an example.
   - **List View**—The results display in a report format. Refer to the *List View* section for more information.

Clearing ERC Markers

Complete the following steps to o clear ERC markers without running a new **Electrical Rules Check**:

1. Select **Tools»Clear ERC Markers** to display the **ERC Marker Deletion Scope** dialog box.
2. Select one of:
   - **Current Page**—To clear the ERC markers from the currently selected page.
   - **Whole Design**—To clear the ERC markers from all pages associated with the design.
3. Click **OK** to delete the selected markers.

ERC Rules Tab

This section describes how to set up the electrical rules used when running an **Electrical Rules Check**. Refer to the *Electrical Rules Checking* section for information about how to run an ERC.

Complete the following steps to set up the electrical rules:

1. Select **Tools»Electrical Rules Check** to display the **Electrical Rules Check** dialog box and click on the **ERC Rules** tab.
2. Set the desired warning or error levels by clicking on the button that appears at the intersection of the desired pin types in the grid found in the **Definition** box. Click until the desired color, based on the **Legend** appears.

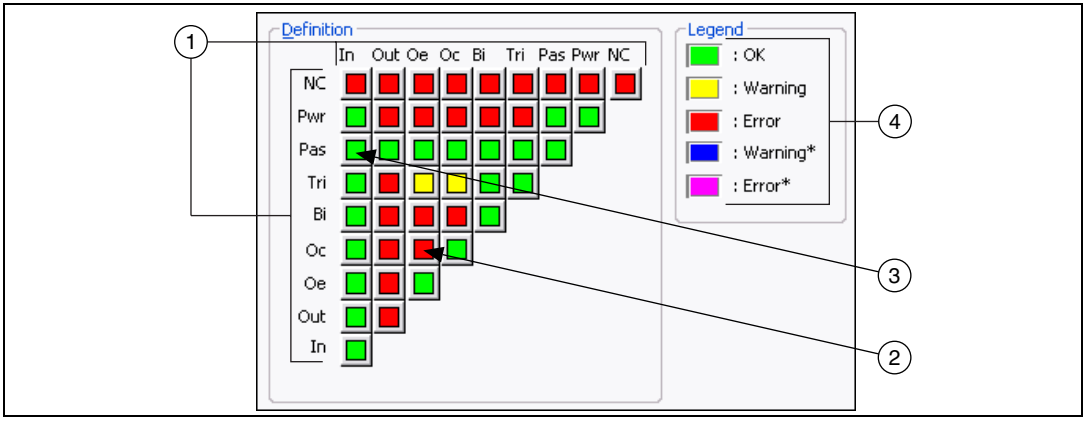
   For clarity, some examples are shown in the figure below.

   The ERC symbols are shown in (1).

   The green button (3) indicates “OK” for connection of Pas (passive) to In (input) pin.

   The red button (2) indicates an error for connection of Oc (open collector) to Oe (open emitter) pin.

   The warning and error levels are shown in the **Legend** area (4). Refer to the *Warning and Error Levels* section for more information.



1	ERC Symbols	3	Green Button
2	Red Button	4	Warning and Errror Levels

Note The table below details the pin types and their associated ERC symbols that are available on the various components in Multisim.

Pin Type	Pin Type from Multisim Component Editor	ERC Symbol
INPUT	Input, 74LS Input, 74S Input, 74 STD Input, CMOS Input, Schmitt Trigger, ECL Input.	In
OUTPUT	Output, Active Driver, 74LS Active Driver, 74S Active Driver, 74STD Active Driver, CMOS Active Driver.	Out
OPEN_COLLECTOR	Open Collector, 74S Open Collector, 74STD Open Collector, CMOS Open Collector, 74LS Open Collector.	Oc
OPEN_EMITTER	ECL Output.	Oe
BI_DIRECTIONAL	Bi-directional, 74LS Bi-directional, 74S Bi-directional, 74STD Bi-directional, CMOS Bi-directional.	Bi
3-STATE	3-state, 74LS 3-state, 74S 3-state, 74STD 3-state, Bi-directional-3st, CMOS 3-State.	Tri
PASSIVE	Passive	Pas
POWER	Power, Vcc, Vdd, Vee, Vpp	Pwr
GND	Gnd, Vss	Pwr
NC	NC (no connection)	NC

Warning and Error Levels

The following explains the warning and error levels available:

- **Ok**—Green button. No message is displayed after an **Electrical Rules Check**.
- **Warning**—Yellow button. A warning message is displayed after an **Electrical Rules Check**.
- **Error**—Red button. An error message is displayed after an **Electrical Rules Check**.
- **Warning***—Blue button. A warning message is displayed after an **Electrical Rules Check**, only if no other pin type is present.
- **Error***—Purple button. An error message is displayed after an **Electrical Rules Check**, only if no other pin type is present.

Component's Pins Tab

Before running an **Electrical Rules Check** you can set which pins to exclude or include for specific components.

Complete the following steps to set up which pins in a component to include or exclude from an ERC:

1. Double-click on the desired component to display its properties dialog box and click on the **Pins** tab.
2. Set the cells in the following columns as desired:
   - **ERC Status**—Select either **Include** or **Exclude** for each of the component's pins.
   - **NC**—Select **Yes** to add NC (no connection) markers to a pin. A pin marked in this way will not have an unconnected pin error assigned to it when you run the ERC. Also, you cannot wire to pins with NC markers. Refer to the *Marking Pins for No Connection* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Note You cannot change the NC state of a pin that is already wired to a net. If you try, a warning appears indicating that this is not permitted.

3. Click **OK** to close the dialog box.

4

Working with Larger Designs

This chapter describes features that are especially useful when dealing with larger, more complicated designs.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Flat Multi-Sheet Design

In many instances circuit designs are too large to fit all components on a single sheet or for logical reasons it's easier to think of a circuit design if it is divided. In this case, you can use Multisim's **Flat Multi-Sheet Design** feature. This allows you to place off-page connectors between different sections of your circuit.

Complete the following steps to add another sheet to a circuit:

1. Select **Place»Multi-Page**. The **Page Name** dialog box displays.
2. Enter the desired name and click **OK**. A blank circuit appears with the name that you entered above.
3. Place components and wire the circuit as desired.
4. Select **Place»Connectors»Off-Page Connector**. A "ghost" image of an off-page connector displays attached to your mouse pointer.
5. Drag the "ghost" image to the desired location and click to place the connector. Repeat for any other required off-page connectors.
6. Wire the off-page connectors into the circuit.
7. Save the file and return to the main circuit window.
8. Select **Place»Connectors»Off-Page Connector**. A "ghost" image of an off-page connector displays attached to your mouse pointer.
9. Drag the "ghost" image to the desired location and click to place the connector. Repeat for any other required off-page connectors.
10. Wire the off-page connectors into the main circuit.

Caution To achieve a connection between a point in the main circuit and a point in another page, the name of the off-page connector in the main circuit must be the same as in the other page. For example, “OffPage1” in the main circuit, will be connected to “OffPage1” in the other (flat) page.

Note Refer to the *Connecting Buses to HB/SCs* section for more information.

Delete Multi-Page Dialog Box

Complete the following steps to delete a page from a multi-page circuit file.

1. Select **Edit»Delete Multi-Page** to display the **Delete Multi-Page** dialog box.
2. Highlight the page that you wish to delete and click **OK**.

Hierarchical Design

Hierarchical blocks and subcircuits are used to organize functionally related parts of a design into manageable pieces. Multisim’s hierarchical functionality allows you to build a hierarchy of inter-connected circuits, increasing the reusability of your circuit designs and ensuring consistency across a group of designers. For example, you might build a library of commonly used circuits, stored in a central location. Those circuits could in turn be contained in other, more complex circuits, which could be used to create yet another level of circuit design. Since the interconnected circuits are linked together, and updated automatically, you can ensure that refinements made to one circuit are carried out in all related circuits as well. This lets you, for example, divide a complex project into smaller, interconnected circuits for completion by individual team members.

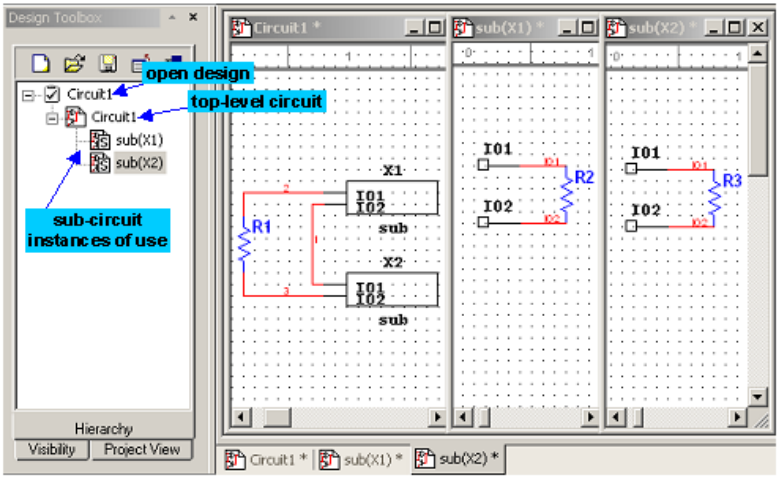
Hierarchical blocks and subcircuits are similar except that subcircuits are saved with the original circuit and hierarchical blocks are individual circuit files that are referenced from a main file. The connection method is the same for both using the **HB/SC Connector**. Sub-circuits are easier to manage, as they cannot accidentally become separated from the circuit that references them. Hierarchical blocks are useful when re-using nested circuits across multiple designs, or for dividing the work when multiple designers are working on the same design.

When using hierarchical blocks, the “block” remains a separate schematic file which can be edited. The connection between a block and the circuit in which it is placed is an active link—if you place the contents of circuit A as a block of circuit B, you can open circuit A separately, make any changes

necessary, and those changes are reflected in circuit B the next time you open it and in any other circuits that use circuit A.

Nested Circuits

When a circuit file is opened or created in Multisim, by definition it is the top-level circuit of the current design. All circuits may reference other, embedded (subcircuit) or linked-to (hierarchical block) nested circuits, which act as building blocks to control circuit complexity. In addition, any circuit (nested or otherwise), may comprise multiple pages for ease of understanding and printing. The **Hierarchy** tab in the **Design Toolbox** displays a graphical view of the open designs.



If the same nested circuit is used more than once in a design, it will appear more than once in the hierarchy view, and will have more than one tab in the main workspace. Each appearance is an *instance of use* of that nested circuit. Use instances are identified by a path formed by the names of the references used to reach them. In the simple example above, there are two ways to reach “sub”: one via the reference X1 in Circuit1 and one via the reference X2.

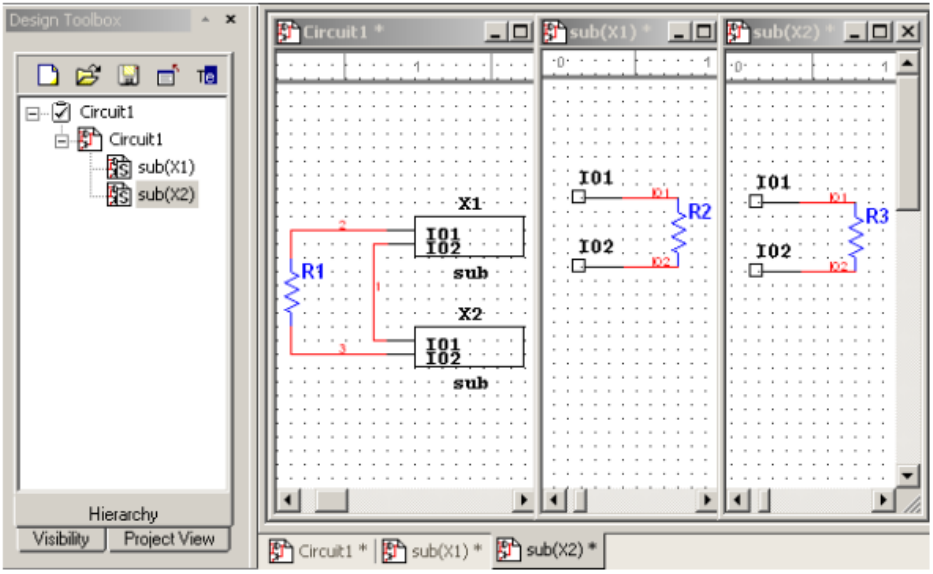
With one notable exception (RefDes assignment), edits made to one instance of use are reflected in all others (because it is actually the same circuit that is being modified). Different views are maintained of each use instance in order that simulation may distinguish them. Thus, if a probe was dragged over the net IO1 in the X1 instance of sub, it would show a different voltage than if it were dragged over the same net in the X2 instance.

Refer to the *Measurement Probe* section of Chapter 9, *Instruments*, for information on probes.

Component Numbering in Nested Circuits

Every part in a design has a unique reference designator (RefDes), such as R5, U2, and so on. A RefDes is assigned by default when a component is placed by taking a single letter typical of the type of part being placed (R for resistors, C for capacitors, and so on) followed by the next highest available number. You may edit these to be anything you like, so long as they are unique across a design. In the case of multi-section parts, the RefDes will also include a section identifier.

The assignment of RefDes to part is stored with the top-level circuit, and not with any of the nested circuits. This is because the same nested circuit may appear more than once in a design. Were the RefDes to be associated with a nested circuit, duplicate RefDes would appear. To combat this, it is the top-level circuit that associates a RefDes with each part's instance of use. Thus, a hierarchical block opened via top-level circuit 1 will have all the same components that are in the same hierarchical block opened via top-level circuit 2, however they will have different RefDes.

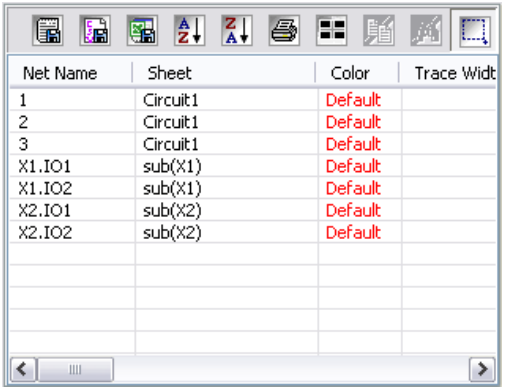


In the above example, we see Circuit1 referencing the same sub-circuit, **sub**, twice. The components, their placement, the wiring, and the net names in the sub-circuit are identical in the two instances (because they are, after

all, the same sub-circuit: **sub**). However, the RefDes of their components are different, as the association of RefDes to component instance is stored with and managed by the containing design, Circuit1.

Net Numbering in Nested Circuits

Net names are unique across all the pages of a multi-page circuit. However, net names may be repeated in nested circuits. This does not cause any ambiguity, however, as the "real" names of nets in nested circuits are formed by pre-pending a dot-separated path of references to reach the circuit instance. In the figure below, 'X1.IO1' is the full name of net number IO1 in the sub-circuit referenced by the RefDes 'X1' from the top-level circuit.



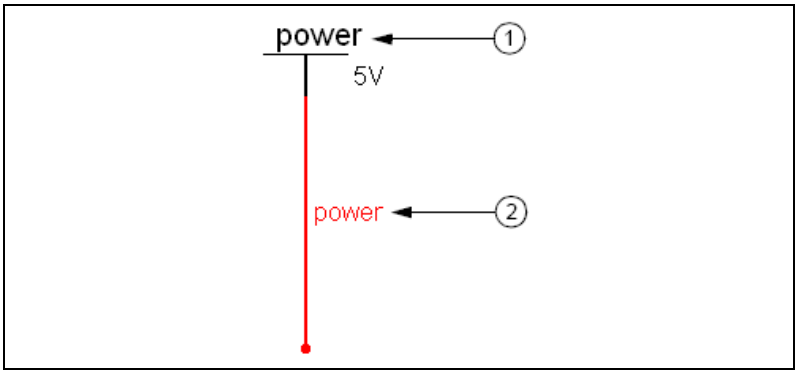
Net Name	Sheet	Color	Trace Widt
1	Circuit1	Default	
2	Circuit1	Default	
3	Circuit1	Default	
X1.IO1	sub(X1)	Default	
X1.IO2	sub(X1)	Default	
X2.IO1	sub(X2)	Default	
X2.IO2	sub(X2)	Default	

Global Nets

Certain pre-defined named nets are global across an entire design—any time a global net at any level in the hierarchy or on any page is re-named to one of these reserved nets, it joins this net. These reserved global nets are 0, GND, VCC, VDD, VEE, and VSS. Net 0 corresponds to analog ground, and is the reference for all voltages during simulation. GND is a digital ground (it is common for the purposes of PCB layout to wish to isolate these two ground nets).

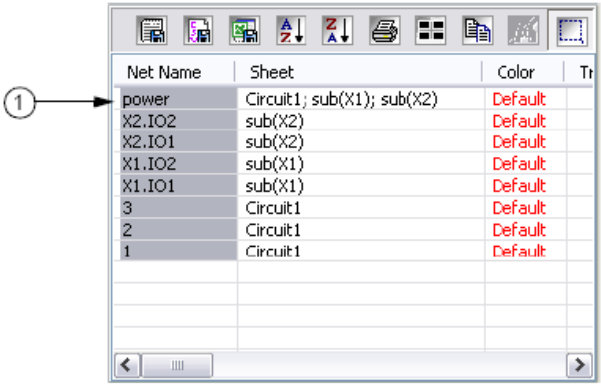
Caution If you want to keep the analog and digital grounds separate during the PCB layout process, be sure that the **Connect digital ground to analog ground** checkbox in the **PCB** tab of the **Sheet Properties** dialog box is *de-selected* before exporting your schematic to Ultiboard. Refer to the *Sheet Properties - PCB Tab* of Chapter 1, *User Interface*, and the *Transferring from Multisim to Ultiboard for PCB Layout* section of Chapter 13, *Transfer/Communication*, for more information.

Components like VCC and VDD that use the “T” symbol, or the GND component that uses the triangle symbol, can be renamed as desired. When a wire is attached to such a component, its net name is automatically changed to the name of the component. For example, in the figure below, VCC was placed and then renamed to “power” by double-clicking on the component and changing the RefDes (1) to “power”. The net name automatically changed to “power” (2) to match the new component RefDes.



1	RefDes	2	Net Name

Nets created this way become accessible across pages in a multi-page circuit, without the use of Off-Page Connectors. If the symbol is placed into a nested circuit (SC or HB), the attached net is renamed to be a top-level net, and joins any other nets at the top-level with the same name. For example, having a net named “power” in a sub-circuit referenced by X1 from the top-level circuit would normally re-write the name of the net to “X1.power”. This makes this net unique to any other net named “power” at the top-level or in any other nested circuit in the design (even other use instances of the same sub-circuit). However, placing a “T” component re-named to “power” and then attaching a wire to it would result in the net being simply called “power”. This creates a virtual wiring situation between all such nets named “power”. Hence, this is a mechanism by which nets may be declared to be of the top-most net namespace, even when buried deeply into nested circuits, as shown in (1) in the following figure.



1 Instances of “power”

Adding a Hierarchical Block

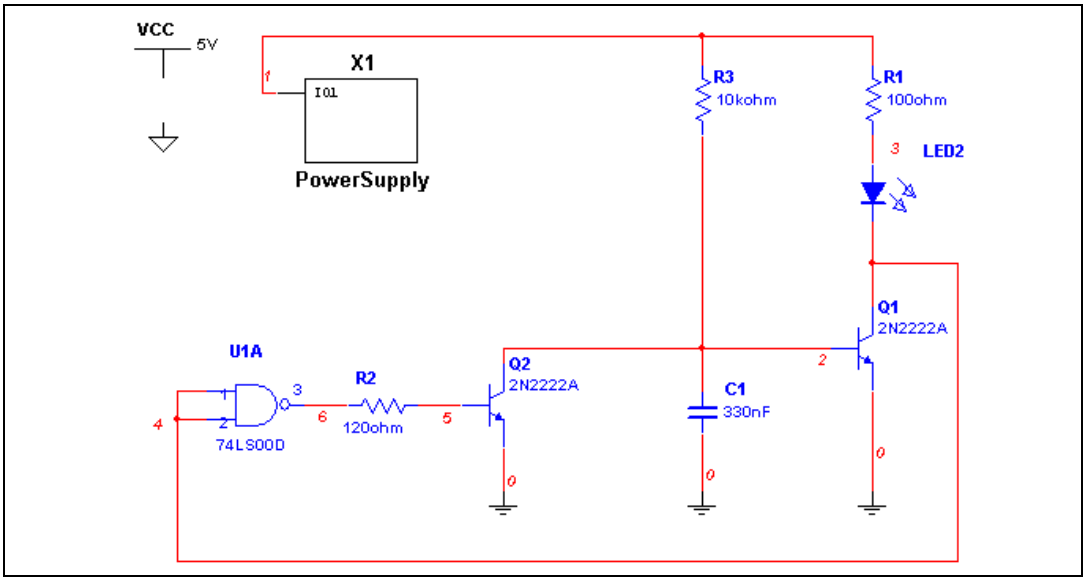
Complete the following steps to place a new hierarchical block:

1. Select **Place»New Hierarchical Block** to display the **Hierarchical Block Properties** dialog box.
2. Enter a filename.

   Or

   Click on **Browse**, navigate to the folder where you would like to save the hierarchical block and click **Save**. You are returned to the **Hierarchical Block Properties** dialog box.
3. Enter the number of pins desired and click **OK**. A “ghost” image of the new hierarchical block appears. Click where you want the hierarchical block to appear.
4. Double-click on the new hierarchical block and select **Edit HB/SC** from the **Label** tab of the **Hierarchical Block/Subcircuit** dialog box that displays. A circuit window that contains only the entered pins displays.
5. Place and wire components as desired in the new hierarchical block.

6. Wire the hierarchical block into the circuit as in the example in the figure below.



7. Save the circuit.

Note If you re-name or move a hierarchical block relative to the main circuit, Multisim will not be able to find it. In this case, a dialog box will display asking you to provide the new location for the hierarchical block.

Placing a HB from an Existing File

Complete the following steps to place a hierarchical block from an existing file:

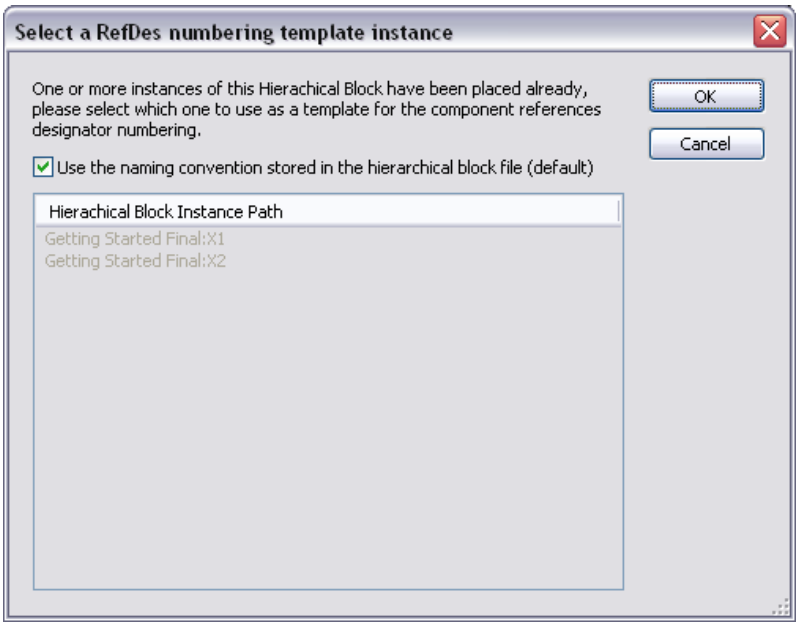
1. Select **Place»Hierarchical Block from File**, navigate to the desired file and click **Open**. The circuit is placed on the workspace.
2. You may need to add HB/SC connectors to the HB if they are not already present.

   To do this, double-click on the placed HB's symbol and select **Edit HB/SC**.

   Select **Place»Connectors»HB/SC Connector**, and place and wire the connector as desired.

   When you return to the main circuit, the symbol for the HB will include pins for the number of connectors that you added.

If you have already placed instances of the hierarchical block, the following dialog displays:


Select a RefDes numbering template instance
One or more instances of this Hierachical Block have been placed already, please select which one to use as a template for the component references designator numbering.
OK
Cancel
Use the naming convention stored in the hierarchical block file (default)
Hierachical Block Instance Path
Getting Started Final:X1
Getting Started Final:X2

3. Select which hierarchical block you wish to use to assign grouping information in the new hierarchical block and click **OK**.

Replacing Components with an HB

Complete the following steps to replace a section of a schematic with a hierarchical block (HB):

1. In the workspace, select the desired components and nets.
2. Select **Place»Replace by Hierarchical Block**. The **Hierarchical Block Properties** dialog box appears. Enter the desired filename and click **OK**.
3. The selection is replaced by a HB symbol that is correctly wired into the circuit.

Adding a Subcircuit

Complete the following steps to place a new subcircuit:

1. Select **Place»New Subcircuit**. The **Subcircuit Name** dialog box appears.
2. Enter the name you wish to use for the subcircuit, for example, "PowerSupply" and click **OK**. Your cursor changes to a "ghost" image of the subcircuit indicating that the subcircuit is ready to be placed.
3. Click on the location in the circuit where you want the subcircuit placed (you can move it later, if necessary).

   The subcircuit appears in the desired location on the circuit window as an icon with the subcircuit name inside it.
4. Double-click on the new subcircuit and select **Edit HB/SC** from the **Label** tab of the **Hierarchical Block/Subcircuit** dialog box that displays. An empty circuit window appears.
5. Place and wire components as desired in the new hierarchical block.
6. Select **Place»Connectors»HB/SC Connector**, and place and wire the connector as desired. Repeat for any other required HB/SC Connectors.

   When you return to the main circuit, the symbol for the subcircuit will include pins for the number of connectors that you added.
7. Wire the subcircuit pins into the main circuit in the usual manner.

Complete the following steps to place another instance of the same subcircuit:

1. Select the desired subcircuit in the workspace and select **Edit»Copy**.
2. Select **Edit»Paste** to place a copy of the subcircuit on the workspace.

Note The copy must be pasted in the same file as the original subcircuit.

Replacing Components with an SC

Complete the following steps to replace a section of a schematic with a subcircuit (SC):

1. In the workspace, select the desired components and nets.
2. Select **Place»Replace by Subcircuit**. The **Subcircuit Name** dialog box appears.
3. Enter the name you wish to use for the subcircuit and click **OK**. The selection is replaced by a SC symbol that is correctly wired into the circuit.

Viewing a Parent Sheet

When you are viewing a subcircuit or hierarchical block you can quickly move to its parent sheet. This is particularly useful when you have many circuits open at once.

To view an active subcircuit or hierarchical block's parent, select **View» Parent Sheet**.

Note This command will move you to the next circuit upwards in the hierarchy. If you have multiple nested circuits and are viewing, for example, a subcircuit within a subcircuit, you will not move to the top of the hierarchy.

Renaming Component Instances

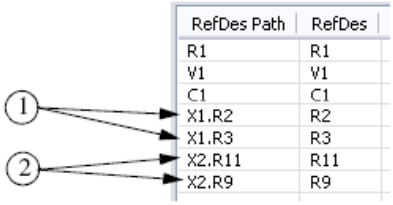
The **Rename Component Reference Designators** dialog box is used to rename/renumber components to, for example, eliminate gaps in numbering. This should be done just before the circuit is exported to PCB layout.

The columns in the dialog box contain the following information:

- **RefDes Path**—The path for the component. If the component is located on the main sheet (that is, *not* in a subcircuit, hierarchical block, or multi-page, the path will only contain the component's reference designator (RefDes), as in the example "V1". If the component is not on the main sheet, the sheet on which it is located also shows, as in the example "X3.R1". This indicates that R1 is found on the subcircuit referenced by X3.
- **RefDes**—The reference designator of the component as it appears on the workspace.
- **Section**—The section of a multi-section component.
- **Locked**—**Yes** indicates that the component will not be changed by either a **Renumber** or a **Gate Optimizer** command.

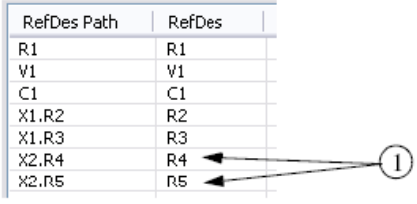
Complete the following steps to renumber the components in a circuit:

1. Select **Tools»Rename/Renumber Components**. The example in the figure below shows the resistors found in subcircuit X1 (1), and the resistors found in subcircuit X2 (2).



1	Resistors in Subcircuit X1	2	Resistors in Subcircuit X2

2. Click **Renumber**. The RefDes's are renumbered in sequence. In the example below, the RefDes's in X2 have been renumbered (1).



1	Renumbered RefDes's in X2

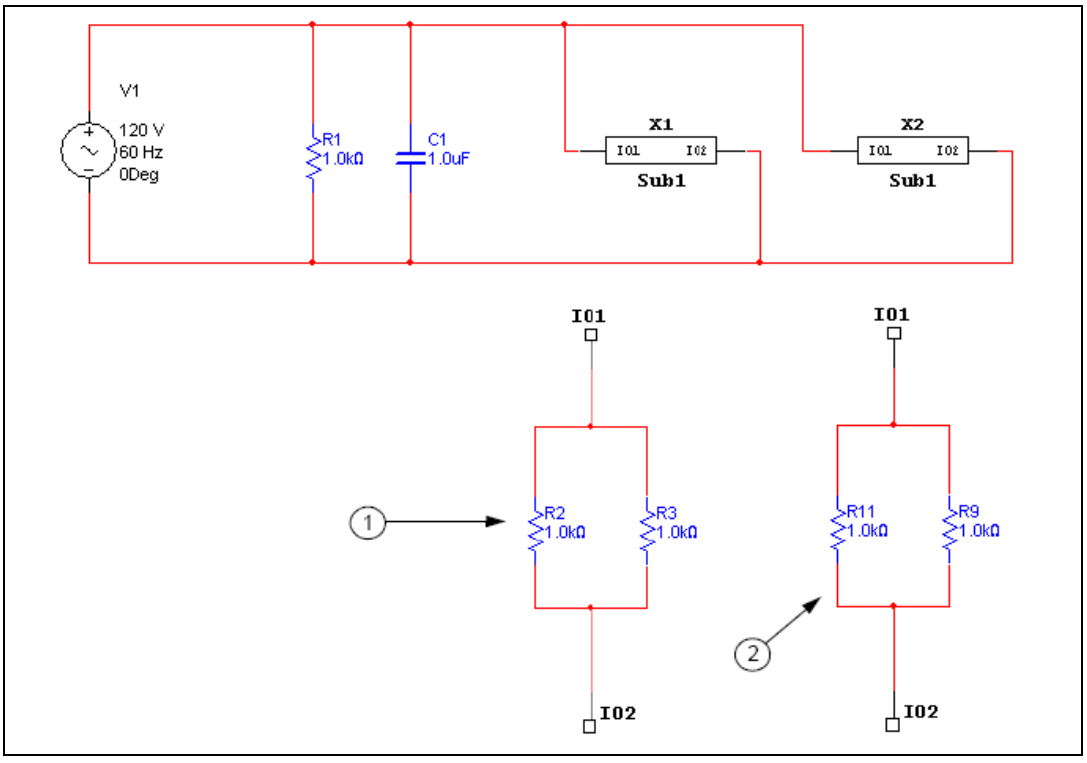
3. Click **OK** to close the dialog box and accept the changes.

Complete the following steps to optimize the circuit so that multi-section components are used as efficiently as possible:

1. Select **Tools»Rename/Renumber Components**.
2. Click **Gate Optimizer**.

Reference Designator Prefix Setup Dialog

The **Reference Designator Prefix Setup** dialog box can be used to assign a unique numbering system for each subcircuit, hierarchical block or multi-page. This is useful when trying to identify components in a large design.This section uses the example of two instances of the same subcircuit as shown in the figure below, (1) and (2).

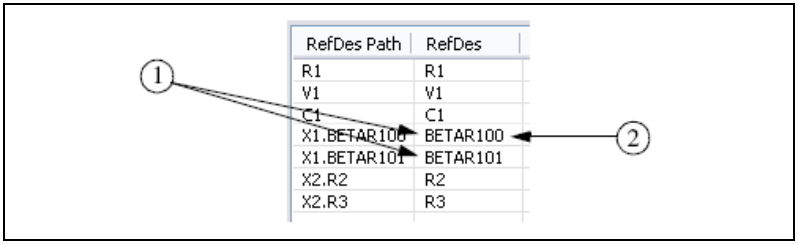


1 X1	2 X2

Complete the following steps to set up renaming parameters:

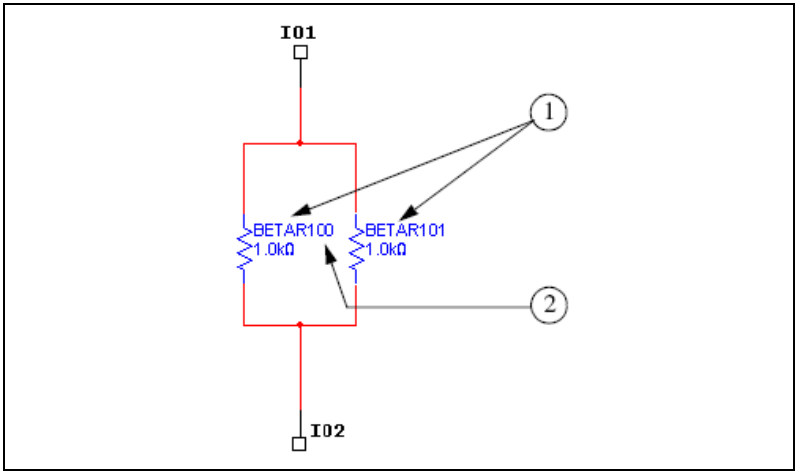
1. Select **Tools»Rename/Renumber Components** to display the **Rename Component Reference Designators** dialog box.
1. Click **Setup**. The **Reference Designator Prefix Setup** dialog box appears.
2. Select the item for which you wish to set up numbering parameters and enable **Use RefDes Prefix**.
3. If desired, enter a prefix for the components in the selection in the **Prefix** field, for example “Beta”.

4. Enter the starting number for each RefDes in the selection in the **Offset** field, for example “100”.
5. Click **OK** to return to the **Rename Component Reference Designators** dialog box.
6. Click **Renumber**. The RefDes’s are renumbered to reflect the changes to the setup that you made in the **Reference Designator Prefix Setup** dialog box. In the example in the figure below, the prefix of “Beta” has been added to each component in the instance of the subcircuit (1). The component numbering sequence for components in the instance of the subcircuit begins at “100” (2).



1	Prefix “Beta”	2	Component Numbering Sequence

7. Click **OK** to accept the changes. In the example in the figure below, the prefix of “Beta” has been added to each component in the instance of the subcircuit (1). The component numbering sequence for components in the instance of the subcircuit begins at “100” (2).



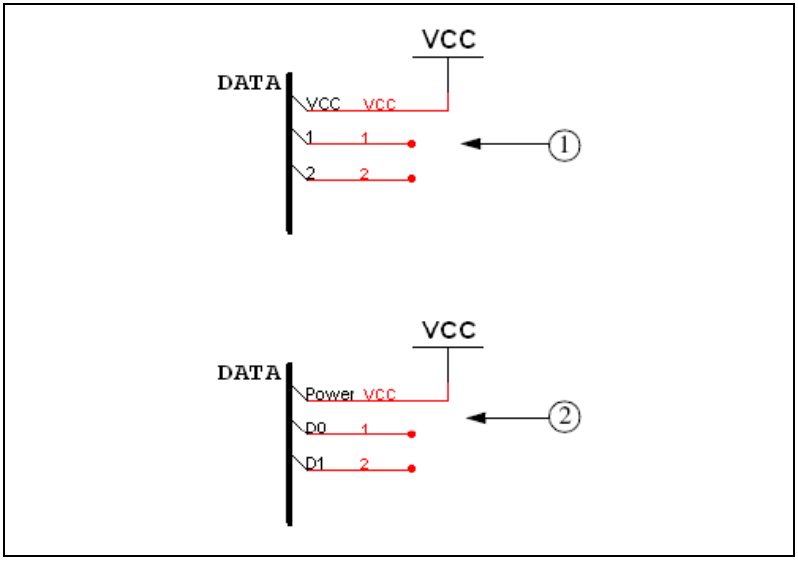
1	Prefix “Beta”	2	Component Numbering Sequence

Buses

In order to simplify wiring, buses can be used to carry multiple nets. Buses can be used within a page, across pages, and down into nested circuits (subcircuits and hierarchical blocks).

Buses operate in two modes. In **net mode**, a bus is simply a collection of nets. Each time a wire is connected to a bus via a bus entry, you are given the choice of connecting that new wire to an existing net already in the bus, or of adding that net to the bus. This is illustrated in the figure below (1).

In **busline mode**, you can pre-define the number and names of buslines that are contained in that bus. When connecting a wire to the bus via a bus entry, you are prompted to specify which of the existing buslines the new wire should attach to. All wires attached to the same busline are merged into the same net. This is illustrated in the figure below (2).

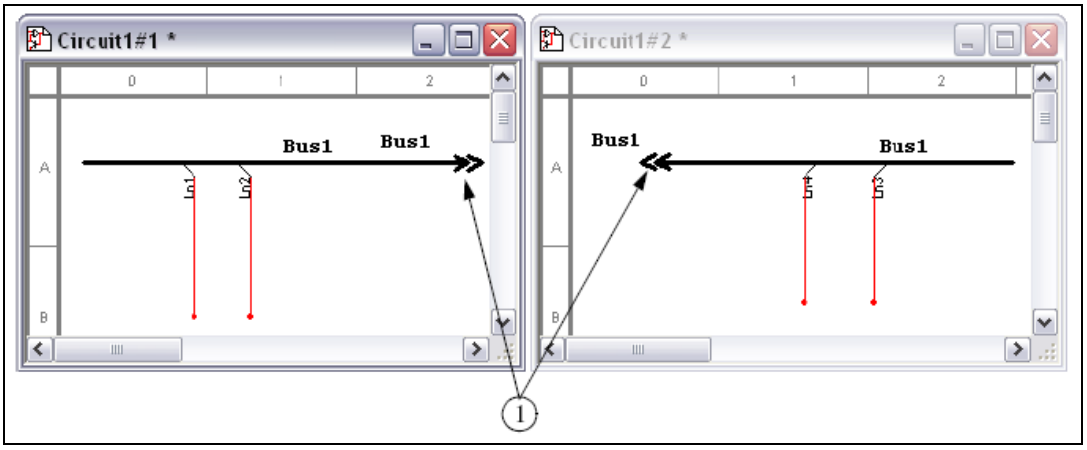


1	Net Mode	2	Busline Mode

Note The bus wiring mode is set in the **Wiring** tab of the **Sheet Properties** dialog box. Refer to the *Sheet Properties - Wiring Tab* section of Chapter 1, *User Interface*, for more information.

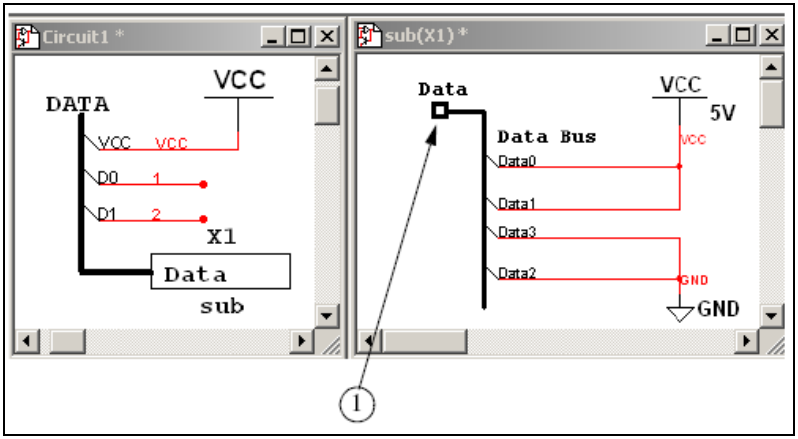
Net mode is a more traditional way of using buses in schematic capture programs. Busline mode allows for a more modern, declaration-before-use, approach to designing buses that minimizes entry errors. Although you can switch between the two modes, you are encouraged to use either one style or the other for any given circuit (switching from busline mode to net mode loses information).

Buses exist in multiple bus segments. All bus segments with the same name are parts of the same bus. They need not all be connected physically. To connect a bus to other pages of the same circuit, a bus off-page connector is used, as in the example shown in the figure below (1). These work analogously to regular off-page connectors in that they allow a bus to be continued onto a second or third page.



1 Bus Off-Page Connectors

A nested circuit can use a Bus HB/SC Connector to specify that one of the pins of the hierarchical block or subcircuit should be a bus pin, as shown in the figure below (1). When using the nested circuit, attaching a bus to such a bus pin brings up a dialog box asking you to map the buslines (or nets) of the bus on the parent circuit to those in the nested circuit. This mapping is necessary as each instance of a nested circuit may in general be connected to a different bus. If the bus in the parent circuit is empty, no dialog will be presented and the bus will be wired in the natural manner.



1 Bus HB/SC Connector

To facilitate the use of buses, a powerful **Bus Vector Connect** facility is provided that allows for multiple pins of a chip to be connected to a bus in a single operation. This dialog allows you to select pins from multi-pin components to connect to the buslines.

Placing a Bus

Complete the following steps to place a bus in your circuit:

1. Select **Place»Bus**.
2. Click on the first point for the bus.
3. Click on the next point for the bus.
4. Continue to click on points until the bus is complete. Buses can be placed horizontally, vertically, and at 45 degrees.
5. Double-click to mark the ending point of the bus.

Placing a Bus across Multi-Pages

Complete the following steps to place the same bus across multi-pages:

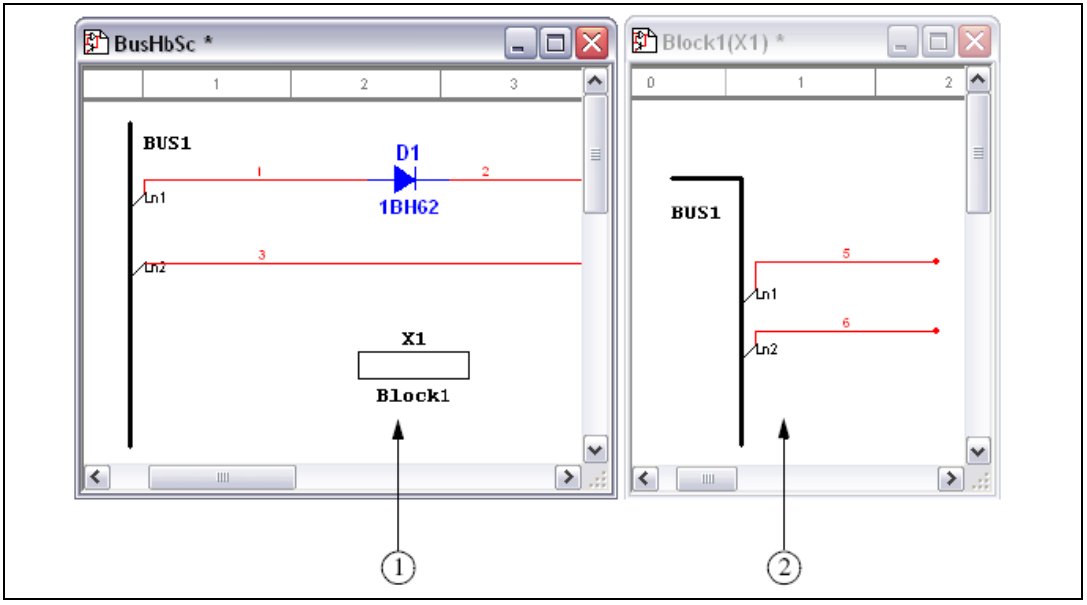
1. Select **Place»Connectors»Bus Off-Page Connector** to place a **Bus Off-Page Connector** on the workspace.
2. Wire the connector to the desired bus.
3. Place a bus on the multi-page, and wire another **Bus Off-Page Connector** to the bus.
4. Double-click on the bus and change its name to match the bus name of the bus on the main page.

Connecting Buses to HB/SCs

Complete the following steps to connect a bus to a bus on a hierarchical block (HB) or a subcircuit (SC):

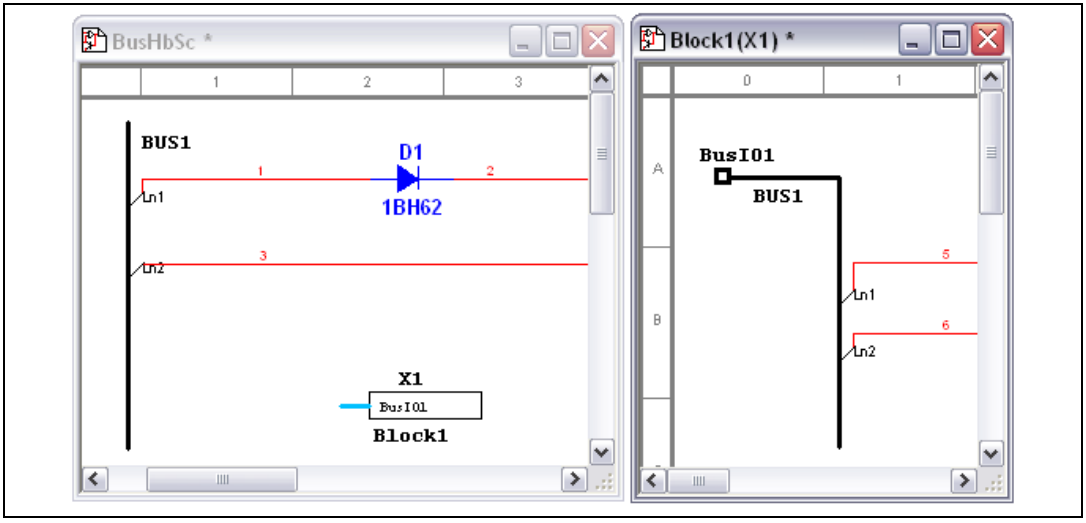
Note This example uses a hierarchical block.

1. Place a bus on the workspace as described in the *Placing a Bus* section.
2. Attach wires to the bus as described in the *Wiring to a Bus* section.
3. Place a hierarchical block or subcircuit on your workspace.
4. Place a bus in the HB/SC and attach wires to it as desired. The example in the figure below shows a hierarchical block in a main circuit (1) and the contents of that hierarchical block (2).



1	Hierarchical Block in Main Circuit	2	Contents of Hierarchical Block

5. Select **Place»Connectors»Bus HB/SC Connector** and place the connector in the HB/SC on the end of the bus. The symbol for the HB/SC in the main circuit will change to reflect the addition of the **Bus HB/SC Connector**, as shown in the example below.



6. In the main circuit, attach the bus to the BusIO pin on the HB/SC by hovering the cursor over *one end of the bus* and clicking when the cursor changes to a crosshair. Move the cursor to the BusIO pin on the HB/SC symbol and click to place. The **Bus HB/SC Mapping Properties** dialog box appears.
7. In the **Bus pin** box, select the buslines from the Bus IO pin you wish to map and click on the activated down-arrow button.
8. In the **Bus** box on the right, select the buslines on the main page to which you wish to map the buslines that you selected above and click the down-arrow.
9. Click **OK**. The connection from the bus in the main circuit is made to the Bus IO pin as mapped above.

Note If you add more nets to the bus, you must double-click on the **Bus IO** pin in the sub-circuit or hierarchical block symbol and map the new nets via the **Bus HB/SC Mapping Properties** dialog box.

Bus Properties

You can add, delete and rename buslines from the **Bus Properties** dialog box.

Adding Buslines to a Bus

When you are in the **busline mode** (refer to the *Buses* section for information) you can pre-enter buslines for use when wiring to a bus.

Complete the following steps to add buslines to a bus:

1. Double-click on a placed bus to display the **Bus Properties** dialog box.
2. Click **Add** to display the **Add Buslines** dialog box.
3. If you wish to add a single busline to the selected bus, enable **Add a Busline** and enter the **Name**.

   If you wish to add a number of buslines, enable **Add Bus Vector** and enter information in the following fields as desired:

   - **Prefix**—The prefix for the entered buslines, for example “Out”.
   - **Start value**—The number from which the buslines will start numbering, for example, “0”.
   - **Increment by**—The size of the step between each number, for example, “1”.
   - **Number**—The total number of buslines to add, for example, “4”.
4. Click **OK**. The **Add Buslines** dialog box disappears, and the **Bus Properties** dialog box appears with the added buslines appearing in the **Buslines (Net)** list.

Deleting Buslines from a Bus

Complete the following steps to delete buslines from a bus:

1. Double-click on a placed bus to display the **Bus Properties** dialog box.
2. Select the desired bus(es) in the **Buslines (Net)** list and click **Delete**.

Renaming Buslines in a Bus

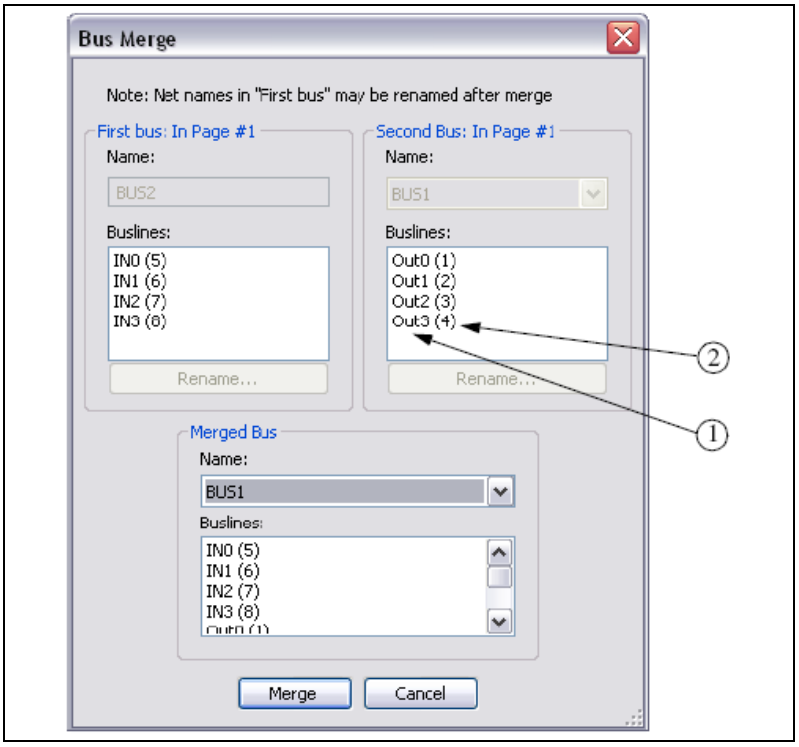
Complete the following steps to rename buslines in a bus:

1. Double-click on a placed bus to display the **Bus Properties** dialog box.
2. Select the desired bus(es) in the **Buslines (Net)** list and click **Rename**. The **Rename Busline** dialog box appears.
3. Enter the desired name in the **New Name** field and click **OK**.

Merging Buses

Complete the following steps to merge two buses together so that they have the same bus name:

1. Highlight the two buses and select **Place»Merge Bus** to display the **Bus Merge** dialog box. Note the example of a busline (1) and the net to which the busline is connected (2) in the figure below.

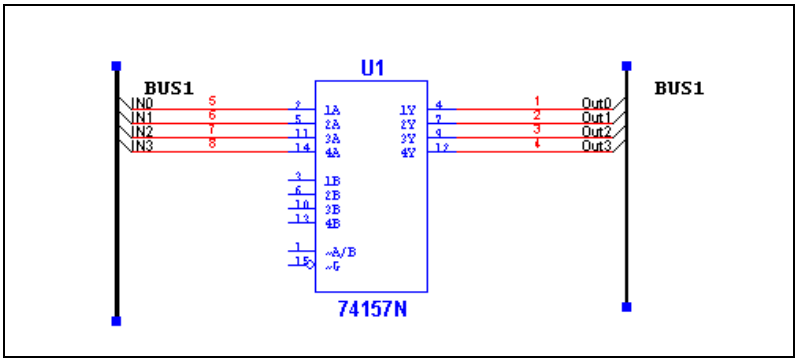


1	Busline	2	Net to Which Busline is Connected

Tip If you want to rename busline(s) before merging the buses, select the desired busline(s) and click **Rename**. Buslines with the same name in each of the merged buses will be electrically connected after the merge.

2. In the **Merged Bus** area, select the bus to use for the merged bus from the **Name** drop-down.

3. Click **Merge**. Note that the two buses on the workspace now share the selected name, as shown in the example in the figure below.

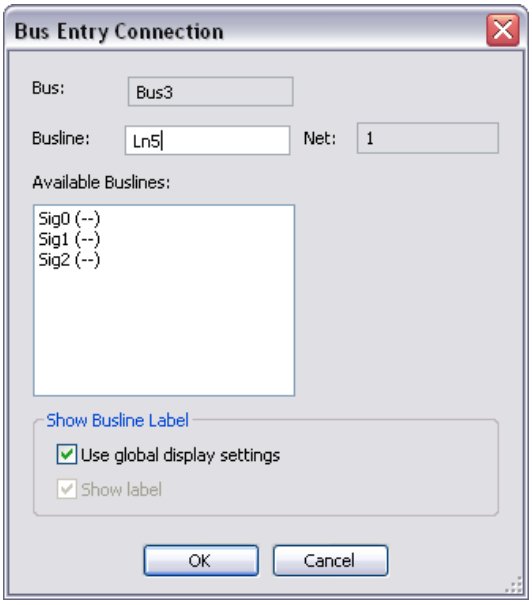


Note You can also access the **Bus Merge** dialog box by double-clicking on a bus, and clicking the **Merge** button in the **Bus Properties** dialog box. In this case, you must select the second bus from its **Name** drop-down list before clicking **Merge**. As well, you can merge buses by wiring them together or renaming one bus with the name of another existing bus.

Wiring to a Bus

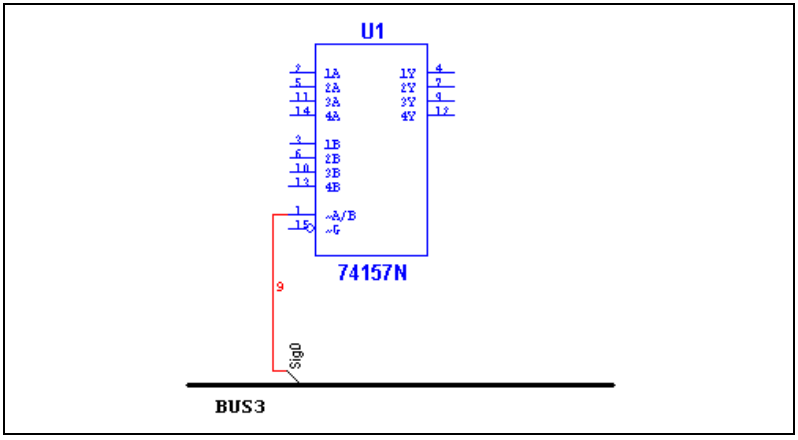
In Busline Bus Wiring Mode

1. Wire the bus into your circuit by drawing a wire to any location on the bus. The **Bus Entry Connection** dialog box appears as in the example in the figure below.



You can use the default **Busline** name or type a new name into the **Busline** field. Or, you can select one of the **Available Buslines**.

2. Select the desired busline and click **OK**.



Tip Make the 45 degree connection, as shown at `Sig0` in the figure above, point in either direction by adjusting the mouse position as you connect the wire to the bus.

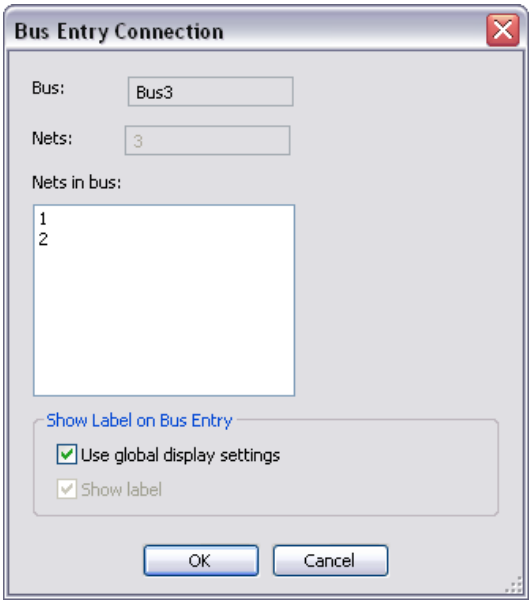
Note After wiring, you can re-name the net, by double-clicking on it and editing the name in the **Net** dialog box that appears.

Note You can select a bus entry to move it with the Arrow keys, or rotate it with <Shift-R>.

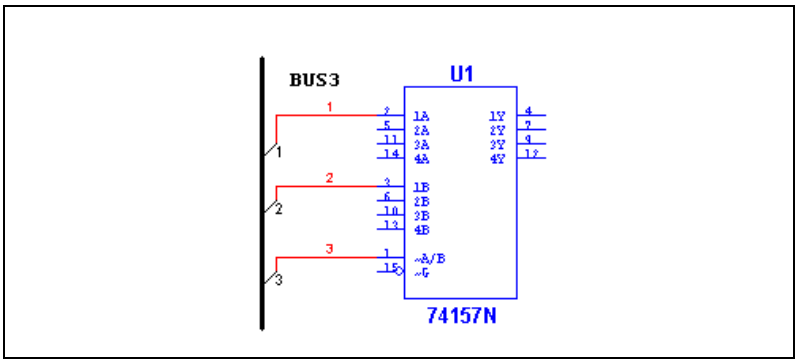
Note You can double-click on a bus entry to edit its properties via the **Bus Entry Connection** dialog box.

In Net Bus Wiring Mode

1. Wire the bus into your circuit by drawing a wire to any location on the bus. The **Bus Entry Connection** dialog box appears:



2. Select the either the default net that appears in the **Nets** field, or select one of the existing nets from the **Nets in bus** list, and click **OK**.



Note After wiring, you can re-name the net as usual, by double-clicking on it and editing the name in the **Net** dialog box that appears.

Note You can select a bus entry to move it with the Arrow keys, or rotate it with `<Shift-R>`.

Note You can double-click on a bus entry to edit its properties via the **Bus Entry Connection** dialog box.

Bus Resizing

To resize a bus, click on the bus and drag one of the handles that appear on the bus.

Complete the following steps to add to the bus:

1. Place the cursor over the end of the bus, and when it turns to a crosshair, click the mouse.
2. Move the cursor to the desired location and double-click to complete the bus.

Bus Vector Connect

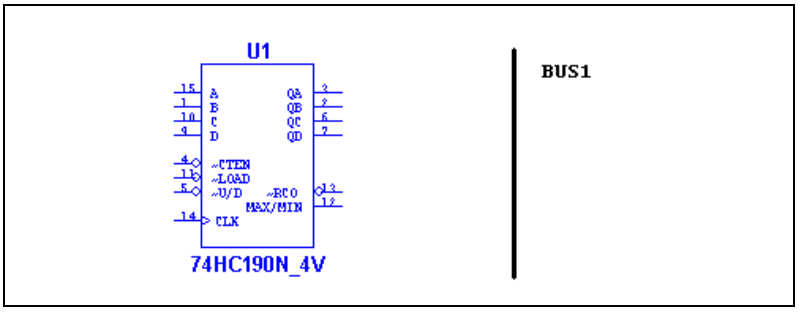
Along with the method described in the *Wiring to a Bus* section, you can also use the **Bus Vector Connect** command. This is the preferred method for placing numerous connections from a multi-pinned device, such as an IC, to a bus.

The following example details the connection of an IC to a bus. However, the **Bus Vector Connect** functionality can be used to connect any multi-pin device to a bus.

In Busline Bus Wiring Mode

Complete the following steps to connect a device to a bus in Busline bus wiring mode:

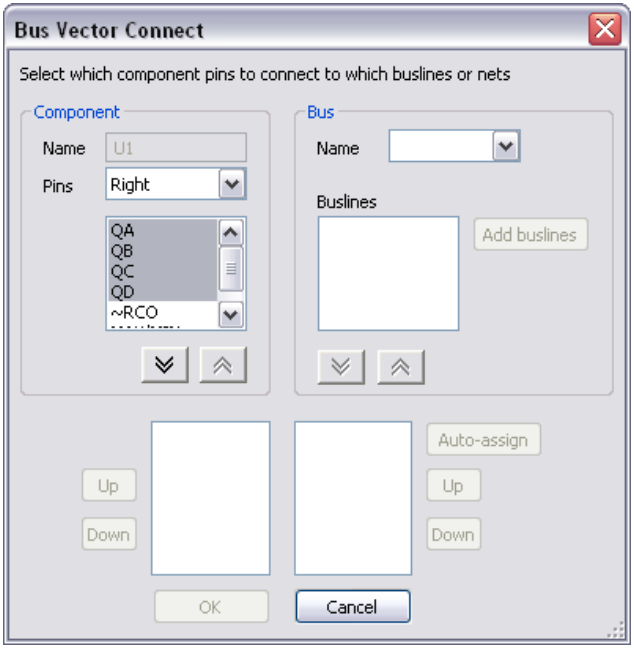
1. Place the component that you wish to connect to the bus on the workspace.
2. Place a bus on the workspace as described in the *Placing a Bus* section.



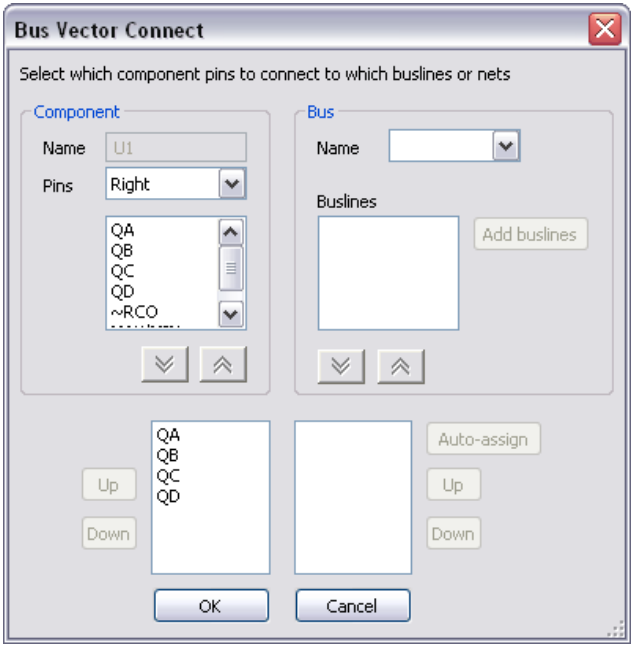
For best results, position the bus so that it is at right-angles to the pins to be connected. The bus should be long enough to comfortably accomodate the number of connections, as in the example in the above figure.

3. Click once on the component to select it and select **Place»Bus Vector Connect** to display the **Bus Vector Connect** dialog box.
4. In the **Component** box, select the side of the component from which terminals should be connected from the **Pins** drop-down list. The list below the **Pins** field is populated based on your selection.

5. Highlight the pins that you wish to attach to the bus, as in the example below.

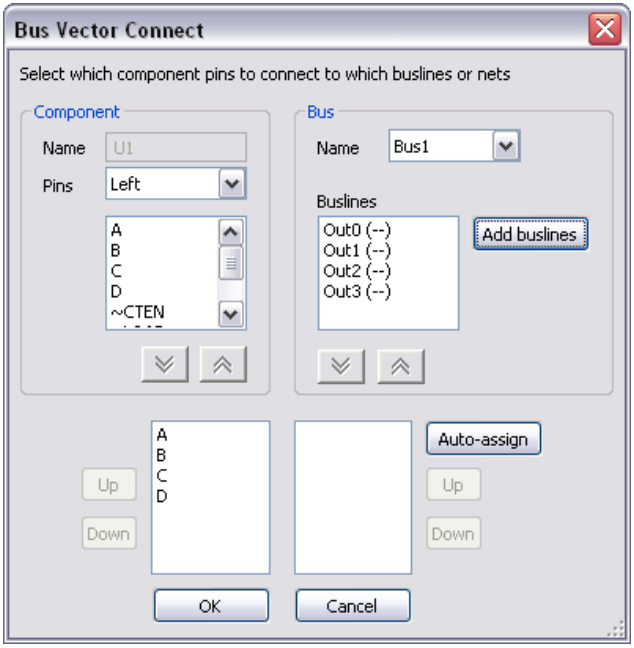


6. Click on the activated down-arrow button to move the selected pins to the bottom left field.

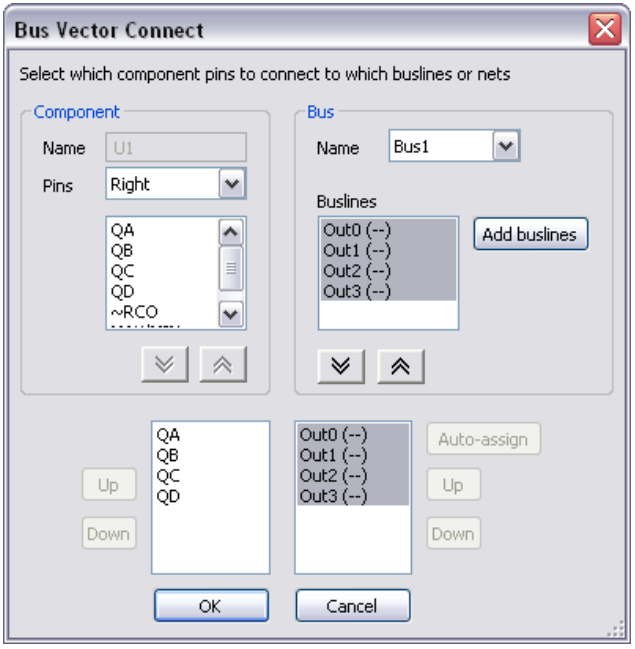


Note If you move an incorrect item, highlight it and click on the up-arrow button to return it to the **Pins** list.

7. In the **Bus** box, select the bus you wish to connect to from the **Name** drop-down list.

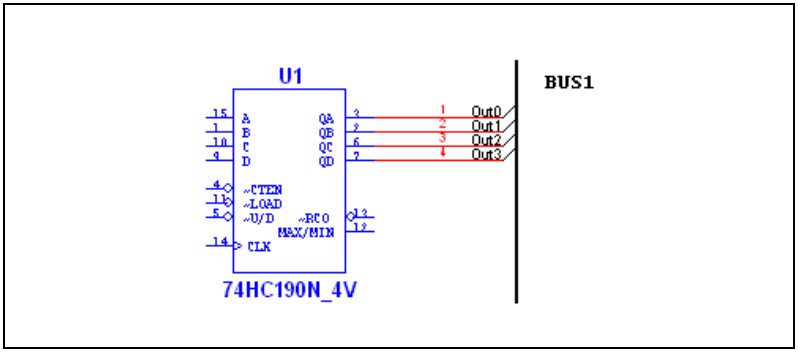


8. In the **Buslines** field, select the buslines you wish to use and click on the enabled down-arrow button to move the selection to the lower-right field.



(If the **Buslines** field is empty, you can click on the **Auto-assign** button to automatically create and assign busline names that correspond to the pin names.)

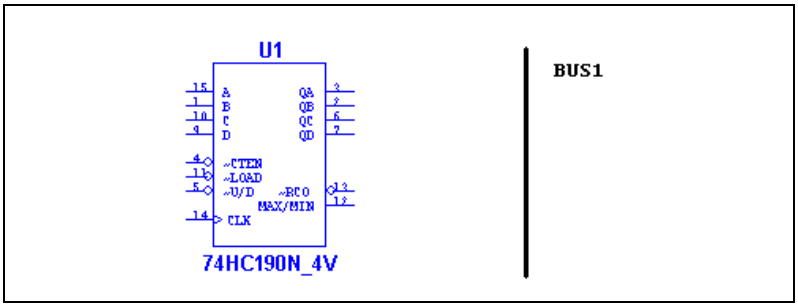
9. Click **OK**. The connections to the bus are made as in the example in the figure below.



In Net Bus Wiring Mode

Complete the following steps to connect a device to a bus in Net bus wiring mode:

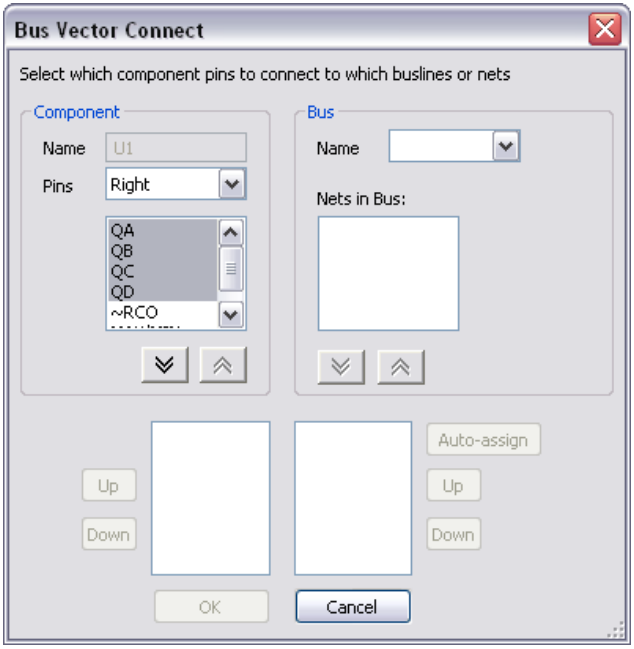
1. Place the component that you wish to connect to the bus on the workspace.
2. Place a bus on the workspace as described in the *Placing a Bus* section.



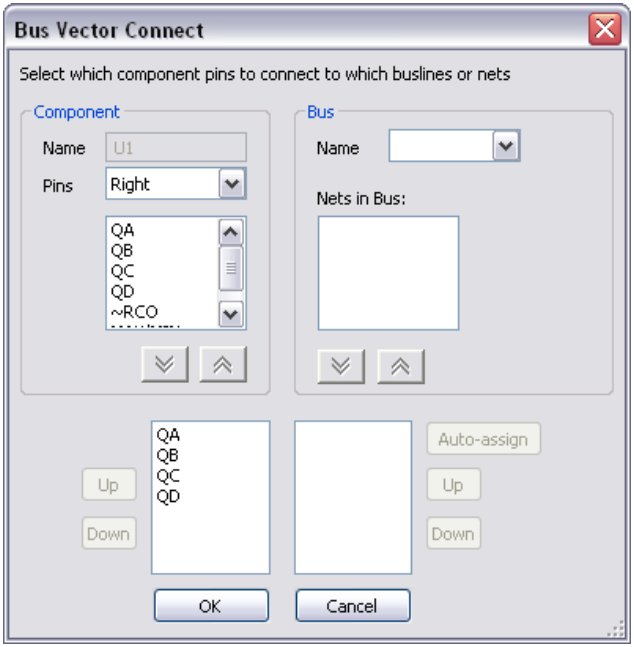
For best results, position the bus so that it is at right-angles to the pins to be connected. Also, the bus should be long enough to comfortably accomodate the number of connections.

3. Click once on the component to select it and select **Place»Bus Vector Connect** to display the **Bus Vector Connect** dialog box.
4. In the **Component** box, select the side of the component from which terminals should be connected from the **Pins** drop-down list. The list below the **Pins** field is populated based on your selection.

5. Highlight the pins that you wish to attach to the bus, as in the example below.

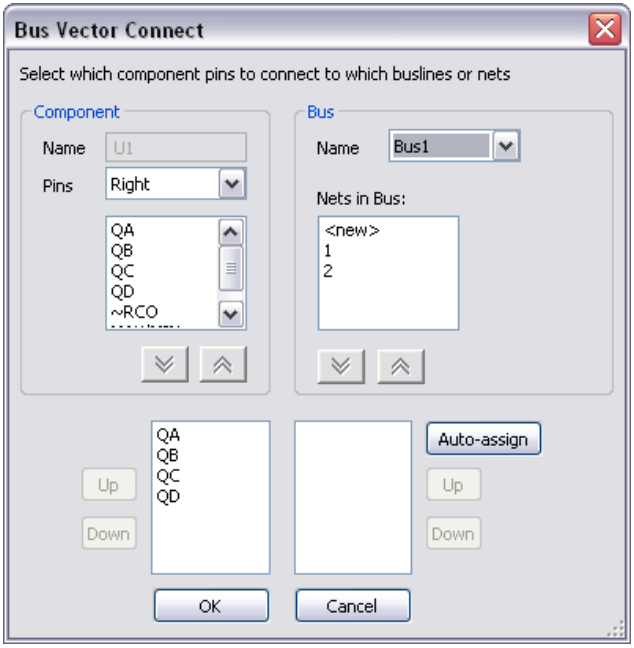


6. Click on the activated down-arrow button to move the selected pins to the bottom left field.

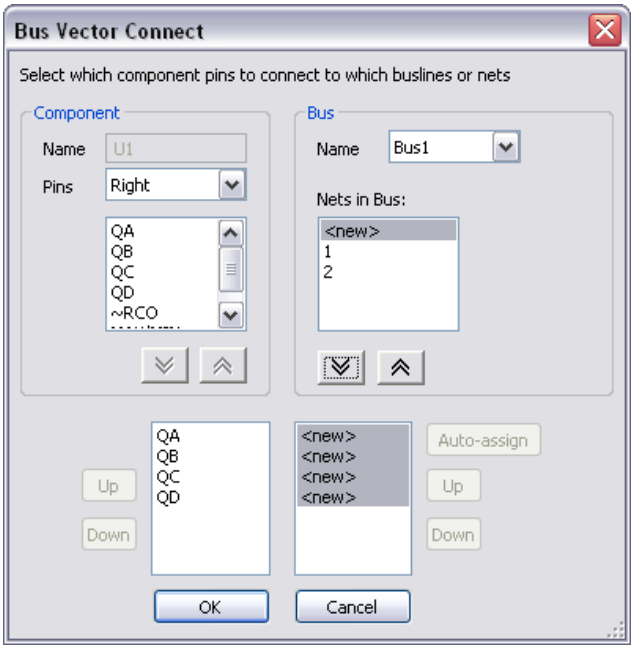


Note If you move an incorrect item, highlight it and click on the up-arrow button to return it to the **Pins** list.

7. In the **Bus** box, select the bus you wish to connect to from the **Name** drop-down list.

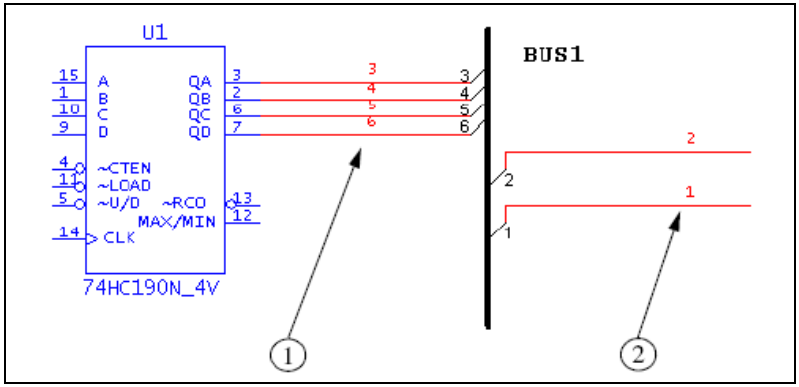


8. In the **Nets in Bus** field select:
   - existing nets—For example, “1” and/or “2” in the above, and click the down-arrow button to move them to the bottom-right field.
   - <new>—To map new nets to the selected component pins. Each time you click the down-arrow button when <new> is highlighted, an instance of <new> appears in the bottom-right field.



Note You can also click **Auto-assign** to automatically assign new nets to the mapped pins.

9. Click **OK**. The connections to the bus are made as in the example in the following figure, which shows the connections made using **Bus Vector Connect** (1), and the previously existing nets (2).



1 Connections Made Using Bus Vector Connect 2 Existing Nets

Variants

A variant is a specific version of a circuit. As PCBs are manufactured for distribution on a global scale, some designs may require modifications depending on their target markets. For example, power supply requirements for the European market differ from those of North America. The variations in the power supply requirements may call for the use of different components in a design. The designer would want to produce a single PCB which would meet requirements for both the North American and the European versions. The board itself must contain the traces as well as land patterns/footprints for both variations of the design. The PCB would then be populated with components according to the target market of the device.

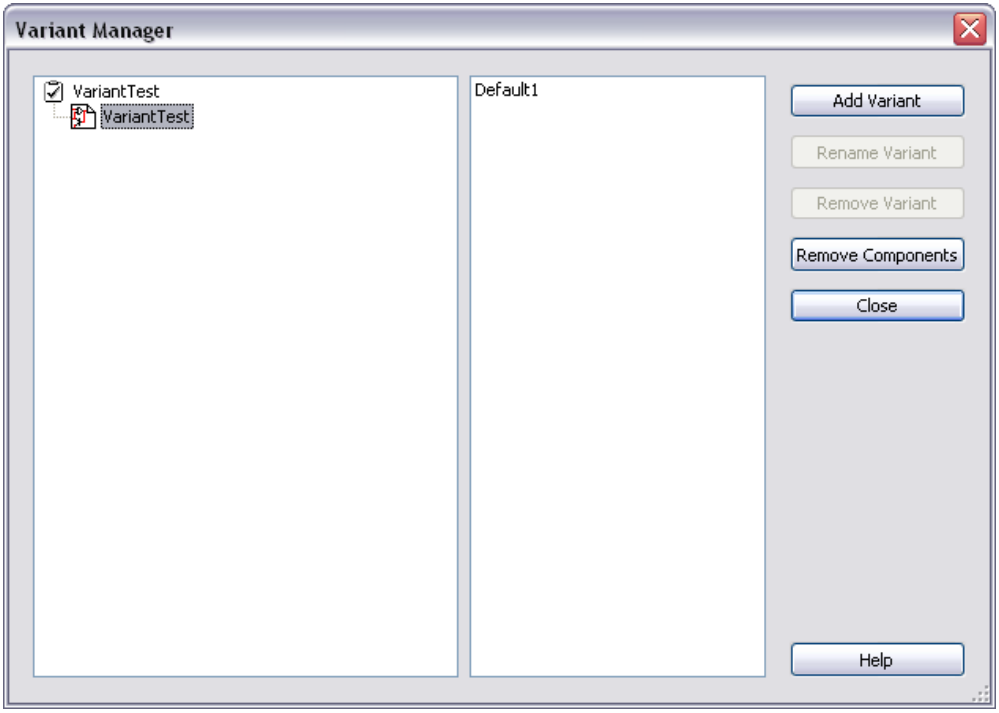
Setting Up Variants

Variants are defined in the **Variant Manager** dialog box.

In the following example variants are entered for North American (NA) and European (EU) versions (that is, variants) of a design.

Complete the following steps to define circuit variants:

1. Open a circuit in Multisim. In this example, the circuit name is "VariantTest".
2. Select **Tools»Variant Manager**. The **Variant Manager** dialog box appears.
3. Highlight the circuit in the hierarchy that is displayed in the left pane. "Default1" is the default name for the initial variant.

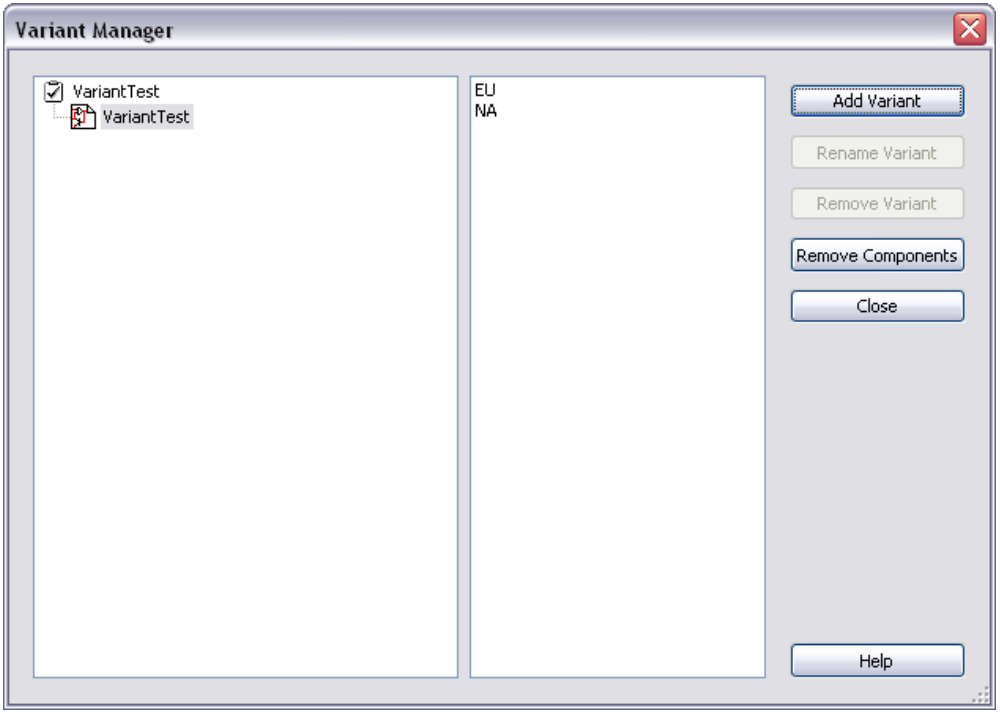


4. Highlight "Default1" in the right pane. The **Rename Variant** button becomes active.

Note The **Remove Variant** button remains disabled, as there must be at least one variable assigned to each circuit. This button becomes active when there are two or more variants assigned to a circuit.

5. Click **Rename Variant**. The **Rename Variant** dialog box appears.
6. Enter a new name for the variant (in this example, "NA", for North America) and click **OK**. The name of the variant changes to the entered name in the right pane of the **Variant Manager** dialog box.

   The name of the European variant must now be entered.
7. Click on **Add Variant** in the **Variant Manager** dialog box. The **Add Variant Name** dialog box appears.
8. Enter the name for the new variant (in this example, "EU" for Europe) and click **OK**. The **Variant Manager** dialog box now appears as shown in the figure below.



9. Click **Close** to return to the workspace.

Complete the following steps to delete variants from your circuit:

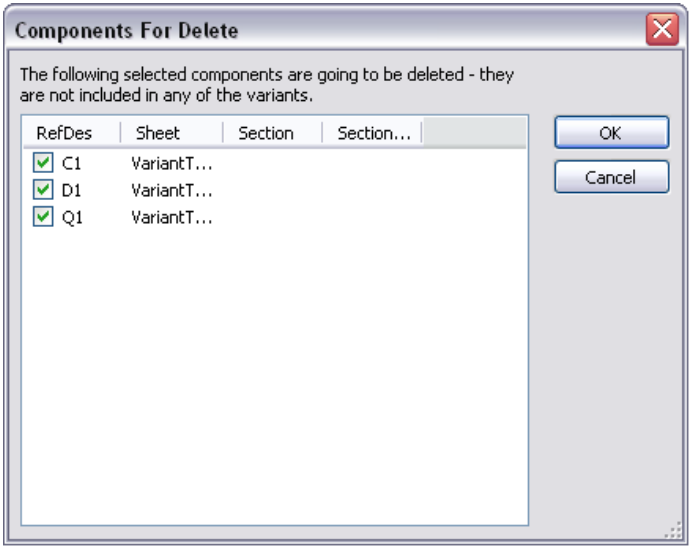
1. Select **Tools»Variant Manager** to display the **Variant Manager** dialog box.
2. Select the desired circuit in the left pane, and the variant you wish to delete in the right pane.
3. Click **Remove Variant**. The highlighted variant is removed.
4. Click **Close** to return to the workspace.

Complete the following steps to rename variants in your circuit:

1. Select **Tools»Variant Manager** to display the **Variant Manager** dialog box.
2. Select the desired circuit in the left pane, and the variant you wish to rename in the right pane.
3. Click **Rename Variant**. The **Rename Variant** dialog box appears.
4. Enter the new variant name and click **OK** to return to the **Variant Manager** dialog box.
5. Click **Close** to return to the workspace.

Complete the following steps to remove components that are not in any of the variants:

1. Click on **Remove Components** in the **Variant Manager** dialog box. The **Components for Delete** dialog box appears.



If you do not wish to remove a component, de-select its checkbox.

2. Click **OK** to remove the components from the workspace.

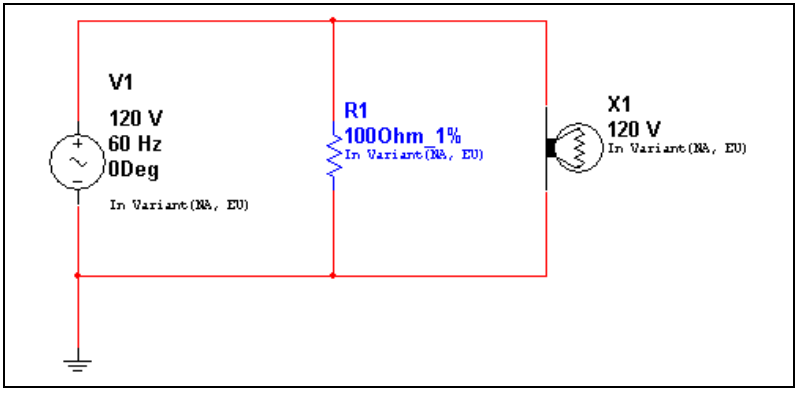
Placing Parts in Variants

This section continues the example of the variants for the North American (NA) and European (EU) markets used in the *Setting Up Variants* section.

Complete the following steps to place a part in a circuit containing variants:

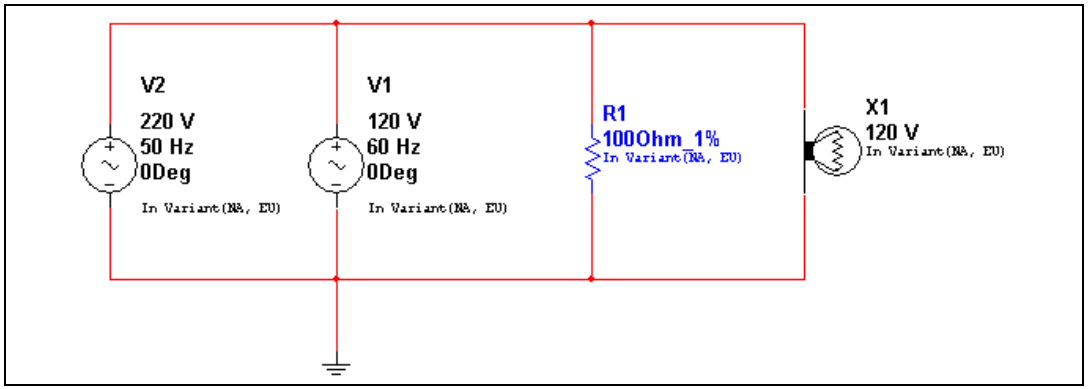
1. Place the parts in the usual manner, as documented in the *Using the Place Component Browser* section of Chapter 2, *Schematic Capture - Basics*.

   After wiring the parts in this example, the circuit appears as follows. Note that at this point, each component is included in both the NA and EU variants of the circuit.

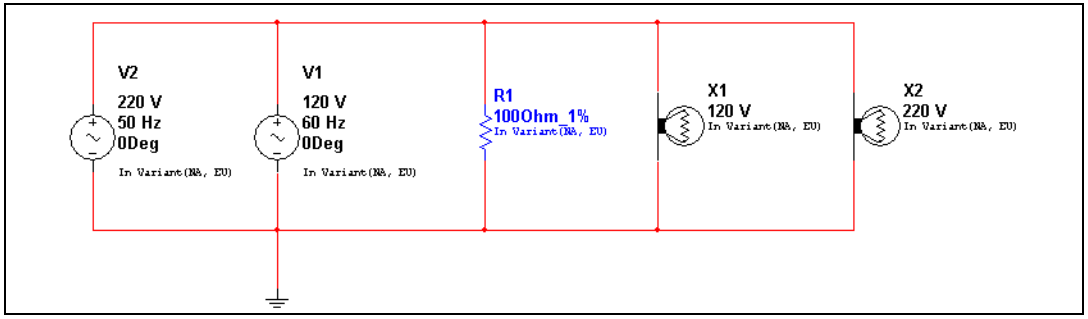


Note To display the variant status for the components, for example, "In Variant (NA, EU)", you must enable the **Variant Data** checkbox in the **Circuit** tab of the **Sheet Properties** dialog box.

2. Add a 220 V, 50 Hz power supply to the circuit. (This is for the EU variant.)



3. Add a 220V lamp to the circuit (also for the EU variant).



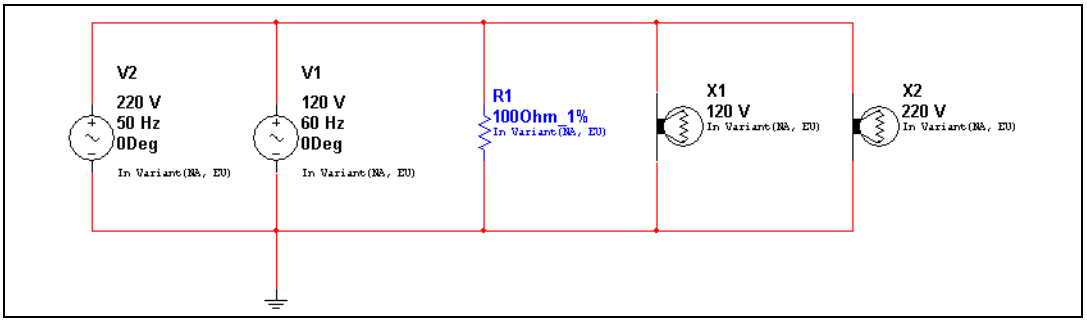
4. Assign variant status (in this case NA or EU) to each component as described in the *Assigning Variant Status to Components* section.

Assigning Variant Status to Components

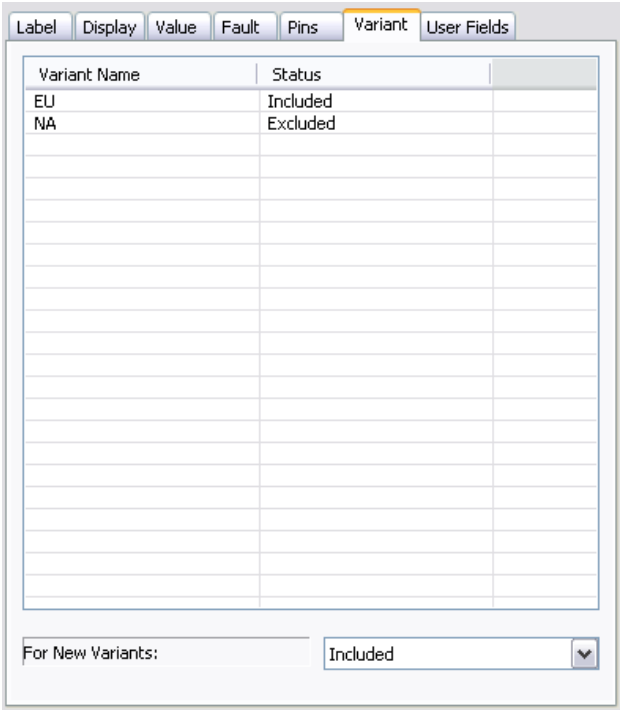
This section continues the example of the variants for the North American (NA) and European (EU) markets used in the *Setting Up Variants* section. After you have set up these variants, you must set which component belongs to which variant.

Complete the following steps to assign components to variants:

1. Open the desired circuit as in the following example:



2. Double-click on a component (for example V2) to display the component's properties dialog box, and click on the **Variant** tab.

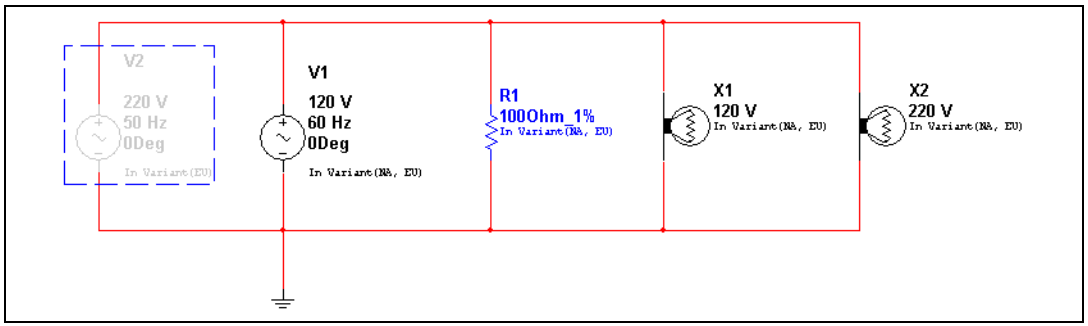


In the above example, **Included** in the **Status** column indicates that the selected component is included in both the EU and NA variants of the circuit.

This is a 220V 50Hz power supply, so we want to include it in the EU (European) variant, but exclude it from the NA (North American) variant.

3. Highlight the line that contains “NA” in the **Variant Name** column and then select **Excluded** from the **Status** column.

4. Click **OK** to close the component’s properties dialog box.



In the above figure, V2 is dimmed, indicating that it is not present in the active variant. Refer to the *Setting the Active Variant for Simulation* section for information on how to set which variant is active on your workspace.

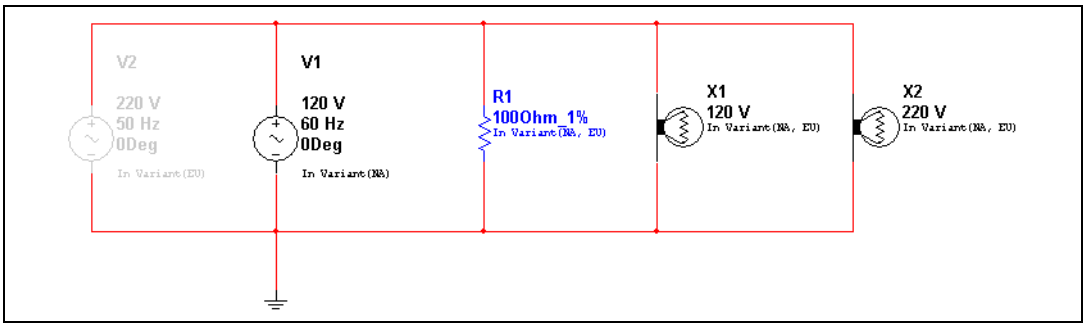
The label in V2 indicates that the component is only in the EU variant (**In EU Variant (EU)**).

5. Continuing with this example, double-click on V1 and select the **Variant** tab.

   This is a 120V 60Hz power supply, so we want to include it in the NA (North American) variant, but exclude it from the EU (European) variant.

6. Highlight the line that contains **EU** in the **Variant Name** column and then select **Excluded** from the **Status** column.

7. Click **OK** to close the component’s properties dialog box.

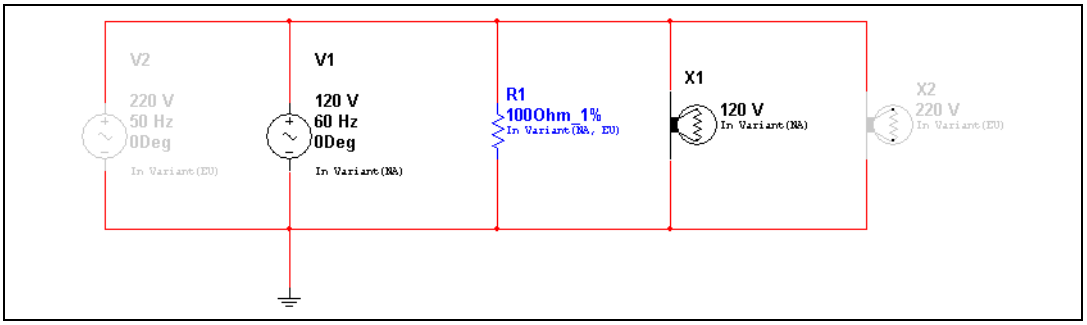


In the above figure, V1 is not dimmed, indicating that it *is* present in the active variant. Refer to the *Setting the Active Variant for Simulation* section for information on how to set which variant is active on your workspace.

The label in V1 indicates that the component is only in the NA variant (**In Variant (NA)**).

8. Set the variant status for the lamps X1 and X2, using the **Variant** tab as described in the preceding steps. X1 is rated at 120 V, so it is for the North American variant, and X2 is rated at 220V, so it is for the European variant.

   When you are done, the circuit will appear as shown below:



Complete the following steps to exclude a component from future variants of a circuit:

1. Double-click on the desired component to display its properties dialog box, and click on the **Variant** tab.
2. Selected **Excluded** from the **For New Variants** drop-down list. This component will not be included in variants of this circuit that you may create in the future.

Note If you create a new variant after setting the status to **Excluded**, and then place an identical component from the database, it will still be included, as that is the default setting for new components.

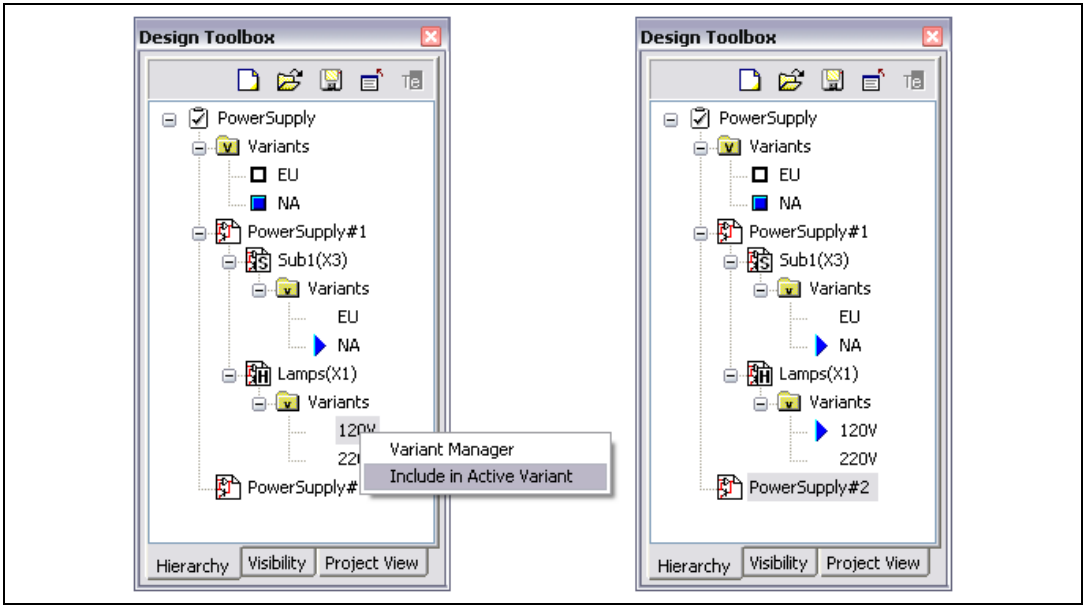
Tip If you copy a component with variants from one circuit and paste it into a circuit with other variants, the pasted component will include the variants from the copied circuit. In this case, you may end up with a circuit that has variants (say USA and North America) that you may wish to combine. You can do this using the **Rename** button in the **Variant Manager** dialog box; just rename one of them to match the other. For example, rename USA to North America and the two variants will be merged into one variant called North America.

Assigning Variant Status to Nested Circuits

The active variant for a nested circuit (HB or SC) is whichever variant is mapped to its parent circuit's active variant. See below for details.

Complete the following steps to assign variant status to a subcircuit or hierarchical block:

1. Right-click on the desired variant, and select **Include in Active Variant** from the pop-up. The active variant is shown by a blue square, as in NA in the example in the figure below. Variants that are included in the active variant are shown by blue triangles, as in 120V as shown in the figure below.



Note You can also assign the variant status in the **Variant** tab of the HB/SC's properties dialog box.

Setting the Active Variant for Simulation

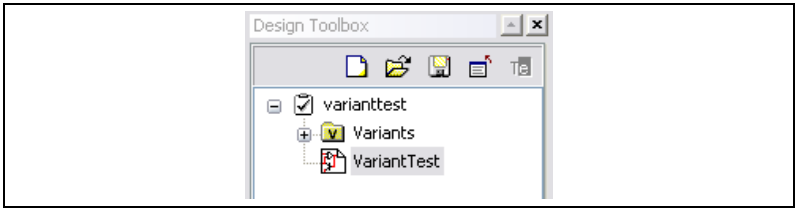
This section continues the example of the variants for the North American (NA) and European (EU) markets used in the *Setting Up Variants* section.

Only one variant of a design can be simulated at a time. Consequently, you must select the variant for simulation from the available circuit variants as described below.

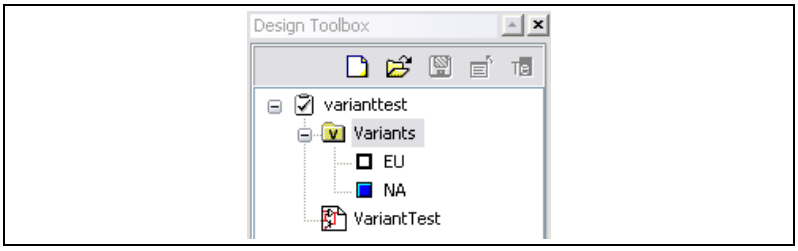
Setting the Active Variant from the Design Toolbox

Complete the following steps to set the active variant from the **Design Toolbox**:

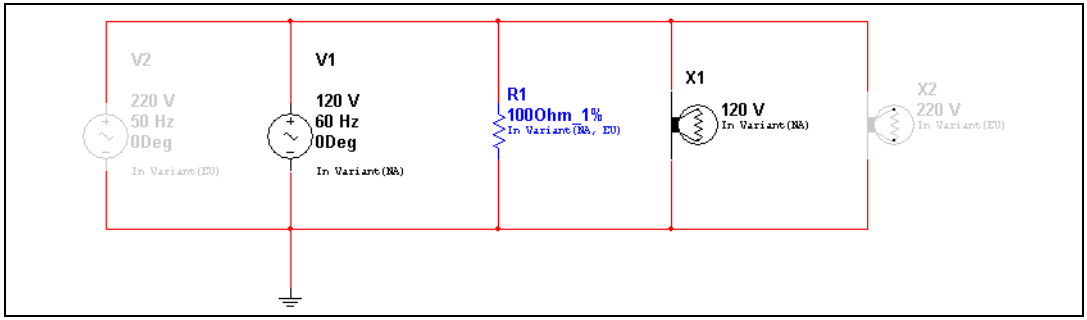
1. Click on the **Hierarchy** tab in the **Design Toolbox**. In the example in the figure below, the hierarchy tree for `VariantTest` is shown.



2. Click on the "**+**" beside the **Variants** folder to open the folder. Active variants are indicated by the blue square, as shown in the example in the figure below (NA); active variants have white squares as in the example below (EU).

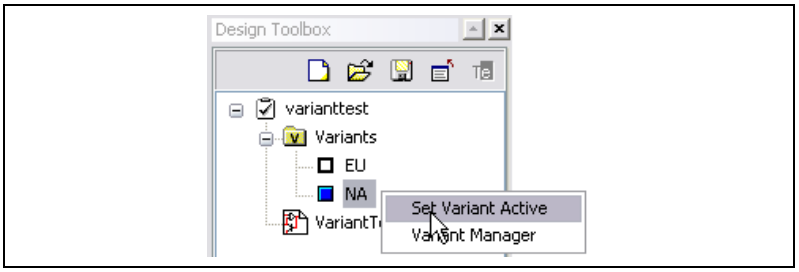


When NA is set as the active variant, the circuit appears as shown in the example below.

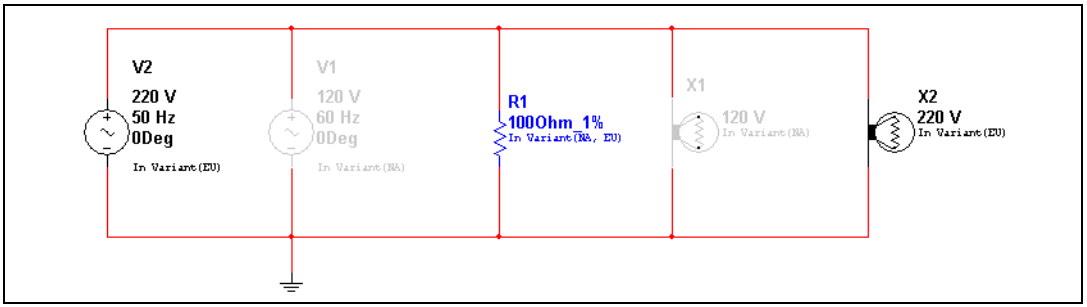


Only the active components will be included when the above circuit is simulated. The dimmed components are inactive, and will not be simulated.

3. Right-click on the EU variant and select **Set Variant Active** from the pop-up that appears, as shown below.



When EU is the active variant, the circuit appears as shown below:



Only the active components will be included when the above circuit is simulated. The dimmed components are inactive, and will not be simulated.

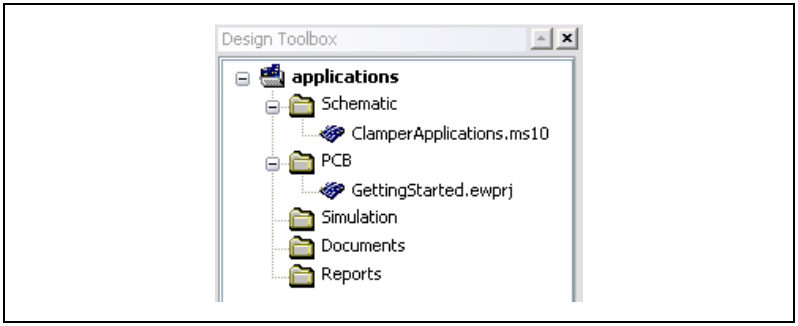
Setting the Active Variant from the Menu

Complete the following steps to set the active variant from the menu:

1. Select **Tools»Set Active Variant**. The **Active variant** dialog box displays.
2. Highlight the variant you wish to make active and click **OK**.

Project Management and Version Control

In order to help manage the various files associated with a design, Multisim **Projects** may be used. A project is a collection of files. For example, all the circuit files making up a design may be grouped together in a project, as well as external design documentation (written in Microsoft Word, for example), simulation output, reports, and PCB layouts generated with Ultiboard. Any file type may be grouped into a project.



Facilities are provided to manage the files in a project as a whole: to version them, back them up, restore them, move them, and lock files within them so that no two designers accidentally work on the same file at the same time.

Note Even if a project contains a top-level circuit that references a hierarchical block, it is not necessarily the case that the hierarchical block is in that same project. This is a choice left entirely up to you. It may not be desirable to backup and version that hierarchical block with the circuit that references it if, for example, the hierarchical block is accessed from many different projects.

Setting up Projects

Complete the following steps to Complete the following steps to create a project for circuit files:

1. Choose **File»New Project**. The **New Project** dialog box appears.
2. Specify the name for your project, the folder where the project files are to be stored, and the folder where backups of the project file are to be placed. If the folders do not already exist, they will be created. Click the button next to the location fields to browse for the desired location.
3. Click **OK**. The **Project View** tab of the **Design Toolbox** appears.

Complete the following steps to add circuit files to the project:

1. Right-click on the **Schematic** folder in the **Project View** tab. Choose **Add file** from the pop-up menu that appears. A standard file selector window appears.
2. Navigate to the location of the circuit file you want included in the project, select it and click **Open**. The file is added to the project and its name appears in the project browser.
3. Hover the cursor over the filename to view its complete filepath.
4. The following folders appear in the **Project View** tab:
   - **Schematic** folder—For circuits drawn as schematics in Multisim. There can be mutiple circuit files in one project, and a circuit file can be part of more than one project.
   - **PCB** folder—For circuits laid out in Ultiboard; are logically the PCBs matching the schematic of the project.
   - **Documents** folder—For documents you wish to collect for the project. For example, MS Word description of project, Excel spreadsheet of costs
   - **Reports** folder—For reports generated by Multisim, for example, **Bill of Materials**, **Netlist Report**.

Complete the following steps to add PCB, document or report files to a project:

1. Right-click on the folder for the desired type of file, for example, **PCB** and select **Add file**.
2. In the file browser that appears, navigate to the desired file and click **Open**.

To remove a file from a project, right-click on the file and choose **Remove**.

To save a project, choose **File»Save Project**.

To close the project, choose **File»Close Project**.

Working with Projects

Complete the following steps to open a file within a project:

1. Right-click on the circuit file name in the **Project View** tab of the **Design Toolbox**.
2. From the pop-up menu that appears, choose either **Open File** or **Open as Read-Only**. If the file is already open by another user, the

Open File command will not be available. If you open the file as read-only, you will not be able to save your changes to that file.

Or

1. Double-click on a circuit file in the project browser. If the file is not in use by another user, it opens. If it is in use, you are prompted to open it as read-only.

Note Files in use by another user are displayed with a different color in the project browser than files that are not in use.

Complete the following steps to open a project:

1. Choose **File»Open Project**. A standard Windows file browser appears.
2. If necessary, navigate to the correct folder and open the project file.

Or

1. Choose **File»Recent Projects** and select the project from the list that appears.
2. Once the project is open, the project browser shows a list of all the files within that project.

Working with Files Contained in Projects

You can lock, unlock, and see summary information about any file in a project.

Complete the following steps to lock the file, preventing anyone else from opening it:

1. Right-click on the file name in the project browser.
2. Choose **Lock File** from the pop-up menu that appears.

Complete the following steps to unlock a file:

1. Right-click on the file name in the project browser.
2. Choose **Unlock File** from the pop-up menu that appears.

Complete the following steps to see information on a file in a project:

1. Right-click on the file name in the project browser.
2. Choose **Properties** from the pop-up menu that appears. The **File Properties** dialog box appears.

Version Control

You can back up the contents of a project folder. You can then restore the folder as of that day and time.

Complete the following steps to back up a project folder:

1. Choose **File»Version Control**. The **Version Control** dialog box appears.
2. Enable **Back up current version**.
3. The system generates a name for the backup, based on the system date. If you wish, you can change this by typing a new name in the field.
4. Click **OK**. The project file is backed up to the filepath shown in the **Location** field.

Complete the following steps to restore a backed up project folder:

1. Close all circuits associated with the project.
2. Choose **File»Version Control**. The **Version Control** dialog box appears.
3. Select **Restore project**. A list of the available backed-up project folders appears at the bottom of the dialog box. (The **Location** field shows where the backup versions of the listed projects are located.)
4. Select the file you want to restore and click **OK**.
5. You are prompted to confirm that you want to over-write the existing project folder contents with the backed up version.

Note Restoring a backed up project folder replaces the current folder. If you want to keep the current folder as well as the backed up version, save the folder to a new location or with a new name before proceeding.

5

Components

This chapter introduces you to the underlying structure and organization of the Multisim component database. It also explains how to access the database for parts and how to search the database for information. Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Structure of the Component Database

The Multisim component database is designed to hold the information necessary to describe any component. It contains everything needed for schematic capture (symbols), simulation (models) and PCB layout (footprints), as well as other electrical information.

There are three levels of database provided by Multisim. The **Master Database** is read only, and contains components supplied by National Instruments. The **User Database** is private to an individual user. It is used for components built by an individual that are not intended to be shared. The **Corporate Database** is used to store custom components that are intended to be shared across an organization. Various database management tools are supplied in order to move components between databases, merge databases, and edit them.

All the databases are divided into groups and then into families within those groups.

When a designer chooses a component from the database and drops it onto the circuit, it is a copy of the component that is placed onto the circuit. Any edits made to the component in the circuit do not affect the original database copy, or any similar components previously placed on the circuit. Likewise, any edits made to the component in the database after a copy has been dropped do not affect the previously placed components, but will affect all subsequently placed ones.

When a circuit is saved, the component information is saved with it. On load, the user has the option to keep the loaded parts as is, to make copies

to place into their user or corporate database, or to update similarly-named components with the latest values from the database.

Database Levels

Components are stored in three different database levels:

- the **Master Database** stores the components as originally shipped with Multisim; to ensure integrity of information, these cannot be edited.
- the **Corporate Database** stores components selected and possibly modified or created by an individual user or company/institution; these remain available to any other selected users.
- the **User Database** stores components modified, imported or created by you; these are available only to you.

The **User Database** and the **Corporate Database** are empty when you first use Multisim. You can use the **User Database** to store frequently used components or components you create using component editing (which is described in the following chapter).

The **Corporate Database** is primarily intended for companies/institutions (or individuals) who work on projects where components with specific attributes are shared within a group or project.

If you modify a component, thereby creating your own version, you must store it in either the **User Database** or **Corporate Database**. You cannot modify the **Master Database** .

You can build a circuit that contains components from any or all of the available databases.

Classification of Components in the Database

Multisim divides components into logical groups. Each group contains families of related components. The groups are listed below:

- Sources
- Basic
- Diodes
- Transistors
- Analog
- TTL
- CMOS

- MCU
- Misc Digital
- Mixed
- Indicators
- Power
- Misc
- RF
- Electro-mechanical
- Ladder Diagrams.

Locating Components in the Database

You can locate components in a specific component family within a specific database by either browsing through the available data, which is the more common method, or by searching for a component that meets specific criteria. These functions are described in this section.

Browsing for Components

When you are placing a component, the browser dialog box that appears lets you browse for components anywhere in the Multisim database. Refer to the *Using the Place Component Browser* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Searching for Components

Multisim comes with a powerful search engine to help you quickly locate components in its database.

Complete the following steps to perform a standard search of the database:

1. Select **Place»Component** to display the **Select a Component** browser.
2. Click **Search**. The **Search Component** dialog box appears.
3. Enter your search criteria (you must enter at least one item) in the appropriate fields. Enter alphanumeric characters, that is, text and/or numbers. Case is not considered.
4. Optionally, click **Advanced Search Using User Fields** to display available user fields, for example, **Vendor**. Enter the desired **Value** for the user field **Title** you wish to search.

5. Click **Search**.

Tip The more specific your search criteria, the smaller the number of matching components.

When the search is complete, the **Refine Search Component Result** dialog box appears, displaying information about the first component that matches your search criteria. The **Component** list contains a list of all the components that match your criteria.

6. From the **Component** list, select the desired component. To view information about any component found by the search, choose it from the list—the displayed fields change accordingly.
7. To place the selected component, click **OK**. You return to the **Select a Component** dialog box, where you can place the component by clicking **OK**.

You can refine your search if your initial attempt yielded a large number of items.

Complete the following steps to refine your search:

1. Click **Refine Search**. The **Refine Search Component** dialog box appears.

Note The original search parameters remain in the **Refine Search Component** dialog until a component is placed.

2. Enter desired parameters and click **Search**.

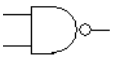
Types of Information Stored for Components

The Multisim databases store information about components in pre-defined fields (that is, fields that are pre-filled in Multisim) and user fields (that is, fields you can use to capture information that you want to record about a component).

Multisim also offers a detailed report of information about components, their models, and their packages. Refer to the *Component Detail Report* section of Chapter 12, *Reports*, for more information.

Pre-Defined Fields

The following pre-defined fields appear in the **Select a Component** browser:

Field	Description	Example
Database	Name of Multisim database in which the component is stored.	User
Group	Name of the group to which the component belongs.	TTL
Family	Name of family to which the component belongs.	74S
Component	Name of the individual component.	74S00D
Symbol	Symbol used (either ANSI or DIN) to represent the component during schematic capture.	
Function	Describes the component. Not available for resistors, inductors or capacitors.	QUAD 2-INPUT NAND
Component type	Describes the construction of resistors, inductors or capacitors only. Contents vary depending on component selection.	Carbon film (for a resistor).
Tolerance	Percent tolerance for the resistor, inductor or capacitor selected in the component list.	0.5
Model Manuf./ID	Name of the company that manufactures the component and the component's ID.	Texas Instruments/74S00
Footprint Manuf./ Type	Footprint for the component (real components only) and the package type. Used in Ultiboard or other vendors' PCB layout products.	DO14
Hyperlink	A link to a desired document.	`www.analog.com`

User Fields

In addition to the fields of data that are pre-defined and filled with information before Multisim is shipped, you can also create your own fields of data to be stored about components. Refer to the *Editing User Fields* section of Chapter 6, *Component Editing*, for information about setting up and entering data into user fields.

Managing the Database

Multisim's database is managed through the **Database Manager** dialog box, which lets you:

- add and remove component families from the **User Database** or **Corporate Database**. You cannot add or remove families in the **Master Database**.
- set up or modify user field titles for any database.
- add or change component toolbar button symbols for component families in the **User Database** or **Corporate Database**.

Complete the following steps to invoke the **Database Manager** dialog box:

1. Click the **Database Manager** button on the **Main** toolbar.

   Or

   Choose **Tools»Database»Database Manager**.

Filtering Displayed Components

Complete the following steps to filter the components that are displayed in the **Components** tab of the **Database Manager** dialog box:

1. Select the desired database from the **Database Name** drop-down list.
2. Click **Filter** to display the **Filters** dialog box.
3. Make selections as described below:
   - **Family** list—Select the desired family. Use the `<Ctrl>` and `<Shift>` keys with the left mouse button to select multiple items in the list.
   - **Component** field—Type the component name. You can also use wildcards.
   - **Show User Data Columns** list—Select the User fields to display.
   - **Select All**—Click to display all User fields.
   - **Clear All**—Click to clear all checkboxes.
4. Click **OK**. The **Filters** dialog box closes and your selections are reflected in the **Components** tab of the **Database Manager** dialog box.

Deleting Components

Complete the following steps to delete an existing component from the **Corporate** or **User** database:

1. Click the **Database Manager** button on the **Main** toolbar.

   Or

   Select **Tools»Database»Database Manager**.
2. Select the **Components** tab.
3. From the **Database Name** drop-down list, choose the database containing the component you want to remove (from the **Corporate Database** or **User Database** only).

Note You cannot delete a component from the **Master Database**.

4. Select the components that you wish to delete. You can use the <Ctrl> and <Shift> keys with the left mouse button to select multiple items in the list.
5. Click **Delete**. You are prompted to confirm the action.
6. Click **Yes**. The components are removed from the database.

Copying Components

Complete the following steps to copy an existing component to the **Corporate Database** or the **User Database**:

1. Click the **Database Manager** button on the **Main** toolbar.

   Or

   Choose **Tools»Database»Database Manager**.
2. Select the **Components** tab in the **Database Manager** dialog box.
3. From the **Database Name** drop-down list, choose the database containing the components you want to copy.
4. Select the components that you wish to copy. You can use the <Ctrl> and <Shift> keys with the left mouse button to select multiple items in the list.
5. Click **Copy**. The **Select Destination Family Name** dialog box appears.
6. In the **Family Tree**, navigate to the database, group and family that you want to copy the component(s) to and click **OK**. You are returned to the **Components** tab in the **Database Manager** dialog box.

Note If there is not a component family in the group that you select, click the **Add Family** button to create one.

7. When you are finished copying components, click **Close**.

Exporting Components

Complete the following steps to export a selected component:

1. Select the desired component in the **Components** tab of the **Database Manager**.
2. Click **Export**. A standard Windows save dialog appears.
3. Navigate to the desired location, and enter the desired **File name**.
4. In the **Save as type** field, select one of:
   - **Component packed file (`*.prz`)**—Select to export all fields in the database for the selected part. (You must import this file into Multisim to read it).
   - **User fields only (Comma delimited) (`*.csv`)**—Select to export the selected component's user fields into a `.csv` file.
   - **User fields only (Tab delimited) (`*.txt`)**—Select to export the selected component's user fields into a `.txt` file.
5. Click **Close**.

Saving Placed Components

If you have made changes to a placed component (for example, changed the footprint) you can save the placed component to either the **User Database** or **Corporate Database**.

Complete the following steps to save a placed component to the database:

1. Select the component on the workspace and select **Tools»Database»Save Component to DB**. The **Select Destination Family Name** dialog box appears.
2. Navigate to the desired component group and family in either the **User Database** or the **Corporate Database**. If necessary, click **Add Family** to create a family in the desired group.
3. Click **OK**. The **Save Components to the Database** dialog box displays with information about the component.
4. Click **OK**.

Moving Components Between Databases

Complete the following steps to move components from the **Corporate Database** to the **User Database**, or vice versa:

1. Choose **Tools»Database»Database Manager**.
2. Select the **Components** tab in the **Database Manager** dialog box.
3. Select the components that you wish to move. You can use the <Ctrl> and <Shift> keys with the left mouse button to select multiple items in the list.
4. Click **Move**. The **Select Destination Family Name** dialog box appears.
5. Navigate to the desired component group and family in either the **User Database** or the **Corporate Database**. If necessary, click **Add Family** to create a family in the desired group.
6. Click **OK** to move the component.

Resetting Master dB User Fields

Complete the following steps to restore the contents of user fields for components in the **Master Database** to their original "factory" settings:

1. Choose **Tools»Database»Database Manager**.
2. Select the **Components** tab in the **Database Manager** dialog box.
3. Select **Master Database** in the **Database Name** drop-down list.
4. Select the desired components and click **Reset User Fields**.

Managing Families

Complete the following steps to add a component family to the **User Database** or **Corporate Database** database:

1. Select the **Family** tab in the **Database Manager** dialog box.
2. In the **Database Family Tree** area, choose either the **Corporate** or **User Database**.
3. Click **Add Family**. The **New Family Name** dialog box displays.
4. Select the desired group from the **Select Family Group** drop-down, type the desired name for the new family in the **Enter Family Name** field, and click **OK**. You are returned to the **Database Manager** dialog box.
5. A default icon is automatically assigned to the new family. To change it, follow the procedure below.
6. The newly-created family appears in the associated component group.

Complete the following steps to load a component toolbar button:

1. With the desired component family selected, click **Load**. You are prompted to navigate to the desired toolbar button file.
2. Select the desired toolbar button file and click **Open**.
3. The new toolbar button will be displayed in the **Database Manager** dialog box in the **Family** area and in the **Family Tree** area under the group where the family was added.

Complete the following steps to edit the default family name button:

1. With the desired component family selected, click **Edit**.
2. Your paint program is launched and the bitmap file of the button is opened.
3. Edit the bitmap file to your requirements and then save and close the paint program.
4. The revised button will appear as the family name button.
5. You can edit both the **ANSI** and **DIN** buttons by selecting the **ANSI** or **DIN** selector in the **Family** area of the **Database Manager**.

Complete the following steps to delete a component family from the **User Database** or **Corporate Database**:

1. Navigate to the component family you wish to delete.
2. Click **Delete Family**. You are prompted to confirm your deletion.
3. The component family is automatically removed.

Complete the following steps to delete empty families from the **User Database** or **Corporate Database**:

1. Click **Delete Empty Families**. You are prompted to confirm your command to delete all empty family folders.
2. To proceed click **Yes**. All empty family folders will be deleted from the **Family** list for the selected database.

Modifying User Field Titles and Content

There are twenty user fields that you can edit to provide user-specific information about the component. Modifying user field titles changes the titles for all databases, not only the selected database.

Complete the following steps to modify user field *titles*:

1. Click the **User Field Titles** tab in the **Database Manager**.
2. Enter the desired names in the **Title** fields.
3. Click **Save**.

Caution Do not change the title of the **Hyperlink** user field, or the links may not function when clicked.

For the majority of non- R, L, or C components in the database, the *contents* of the user fields are entered and edited from the **Components** tab of the **Database Manager**. Refer to the *Modifying User Field Content for RLC Components* section for information about resistors, inductors, or capacitors.

Complete the following steps to modify user field contents for non-RLC components:

1. Click the **Components** tab in the **Database Manager**.
2. Select the desired **Database Name** and component.
3. Right-scroll to the desired user field for the selected component.
4. Enter the desired information in the user field.

Caution User field content for **Master** dB components is stored in the **Corporate** database. Consequently, if the **Corporate** dB is deleted, the contents of their user fields no longer appear in the **Master** database.

Modifying User Field Content for RLC Components

Complete the following steps to modify user field contents for RLC components:

1. Click the **RLC Components** tab in the **Database Manager**.
2. Select the desired component.
3. Right-scroll to the desired user field for the selected component.
4. Enter the desired information in the user field.

Caution R, L, and C components found in the **Master** database are in fact a combination of the component's value, component type, tolerance, and footprint manufacturer/type. Whenever a unique combination of these values is selected in the **Select a Component** dialog box, and the **Save unique component on placement** checkbox is enabled, the combination of these values is saved in the **Corporate** database. Consequently, if the **Corporate** dB is deleted, these components no longer appear in the **Master** database. Any user field information is also lost.

Displaying Database Information

Complete the following steps to review database information:

1. Click the **About** button in the **Database Manager** dialog box. The **Database Information** dialog box displays.
2. To view version and other information, scroll down as desired.

Editing Components

Refer to Chapter 6, *Component Editing*, for detailed information about editing components.

Converting Databases

If you are a user of an earlier version of Multisim, your **User Database** and **Corporate Database** must be converted to Multisim 10 format if you wish to use their components in Multisim 10.

Note The option to convert the **Master Database** is not available, as a new **Master Database** is loaded when you install Multisim.

Complete the following steps to update your databases:

1. Select **Tools»Database»Convert Database**. The **Convert Database** dialog box appears.
2. In the **Type** drop-down list select one of:
   - **Convert DB V8/V9 ==> V10**—Converts V8 or V9 components to V10 format.
   - **Convert DB V7 ==> V10**—Converts V7 components to V10 format.
   - **Convert DB V6 ==> V10**—Converts V6 (Multisim 2001) components to V10 format.

   The title bar of the dialog box changes to reflect your selection.
3. Click **Select Source Database Names**. The **Select a Component Database Name** dialog box displays.
4. Select the type of database that you wish to convert from the **Files of type** drop-down list:
   - **User**—User database.
   - **Corporate**—Corporate database.

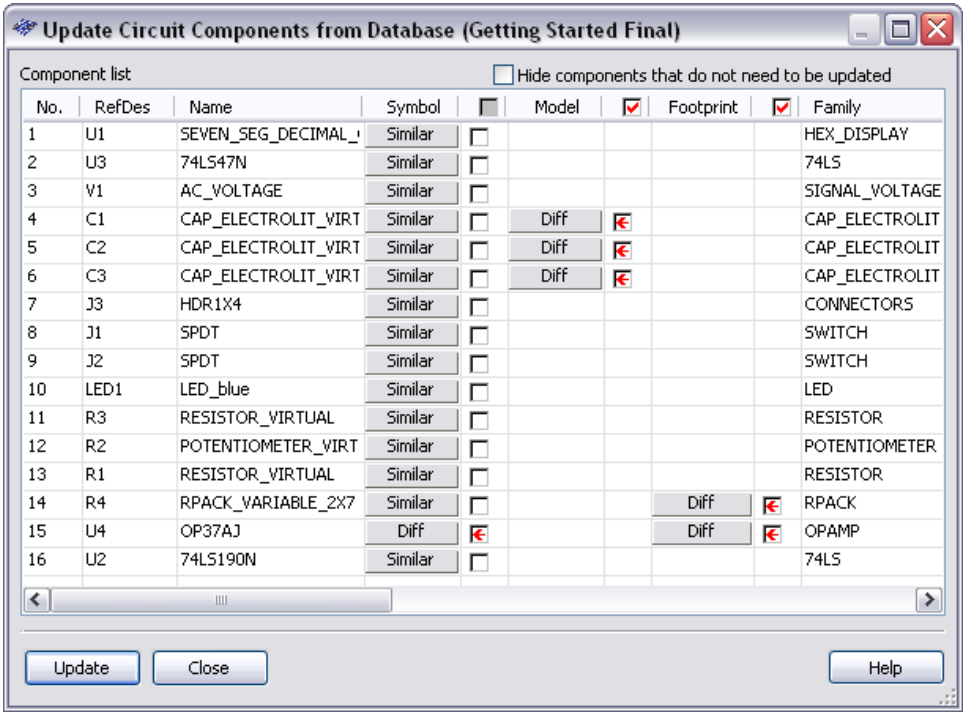
5. Highlight the desired database file (the one you wish to convert) and click **Open**. You are returned to the **Convert Database** dialog box.
6. Click **Start**. The **Duplicate Component Name** dialog box displays.
7. Select the desired option and click **OK**. The database is converted.
   - **Auto-Rename**—Import and automatically rename the duplicate components.
   - **Overwrite**—Replace the Multisim components with the older components.
   - **Ignore**—Do not import components with duplicate names.
8. Click **Close** to close the **Convert Database** dialog box.

Updating Components from Databases

If you open a circuit that was created in an older version of Multisim, you should update its components to match the current database by following the procedure below.

Complete the following steps to update components:

1. Select **Tools»Update Circuit Components**. A dialog box similar to the following displays.


Update Circuit Components from Database (Getting Started Final)
Component list
Hide components that do not need to be updated
No. RefDes Name Symbol Model Footprint Family
1 U1 SEVEN_SEG_DECIMAL_ Similar HEX_DISPLAY
2 U3 74LS47N Similar 74LS
3 V1 AC_VOLTAGE Similar SIGNAL_VOLTAGE
4 C1 CAP_ELECTROLIT_VIRT Similar Diff CAP_ELECTROLIT
5 C2 CAP_ELECTROLIT_VIRT Similar Diff CAP_ELECTROLIT
6 C3 CAP_ELECTROLIT_VIRT Similar Diff CAP_ELECTROLIT
7 J3 HDR1X4 Similar CONNECTORS
8 J1 SPDT Similar SWITCH
9 J2 SPDT Similar SWITCH
10 LED1 LED_blue Similar LED
11 R3 RESISTOR_VIRTUAL Similar RESISTOR
12 R2 POTENTIOMETER_VIRT Similar POTENTIOMETER
13 R1 RESISTOR_VIRTUAL Similar RESISTOR
14 R4 RPACK_VARIABLE_2X7 Similar Diff RPACK
15 U4 OP37AJ Diff Diff OPAMP
16 U2 74LS190N Similar 74LS
Update
Close
Help

If there are any differences, they will be indicated by a **Diff** button in the appropriate column and a red arrow in a box to its right. For information about similar components, refer to the *Updating Similar Component Symbols* section.

2. Optionally:
   - If a **Diff** button appears in the **Symbol** column, click on it to show the difference in symbols between the component on the workspace and the component in the current database.
   - If a **Diff** button appears in the **Model** column, click on it to show the difference in models between the component on the workspace and the component in the current database.
   - If a **Diff** button appears in the **Footprint** column, click on it show the difference in footprints between the component on the workspace and the component in the current database.
3. Select the items to update as follows:
   - **Symbol** column—Be sure that all components with a **Diff** button in this column have their corresponding "red arrow" box enabled. For example, U4 in the figure in step 1.
   - **Model** column—Be sure that all components with a **Diff** button in this column have their corresponding "red arrow" box enabled.
   - **Footprint** column—Be sure that all components with a **Diff** button in this column have their corresponding "red arrow" box enabled.

Note To update all items in a **Symbol**, **Model**, or **Footprint** column, enable the **Update** checkbox to the right of the colum title.

4. Click **Update** to update the selected components.

Updating Similar Component Symbols

Components with a **Similar** button in the **Symbol** column do not need to be updated for the circuit to simulate correctly, or for the circuit to be exported to PCB layout software without error.

Consequently, you may wish to select the **Hide components that do not need to be updated** checkbox. If you choose not to enable this checkbox, you can click any **Similar** button that appears, and determine whether or not to update it based on the information that appears. If you wish to update a similar **Symbol**, be sure to enable its corresponding "red arrow" box before you click the **Update** button in the dialog box.

Merging Databases

Complete the following steps to merge the contents of another database into your **User** or **Corporate Database**:

1. Select **Tools»Database»Merge Database**. The **Database Merge** dialog box appears.
2. Click **Select a Component Database Name** and navigate to the database that you wish to merge into your database and click **Open**. You are returned to the **Database Merge** dialog box.
3. Select the desired **Target Database**.
4. Click **Start**. The selected database is merged into your **Corporate** or **User Database**.
5. Click **Close**.

6

Component Editing

This chapter explains how to create, edit, copy, or delete a component. It also explains how to load any models into the Multisim database that you may have developed, obtained or purchased, and how to create simulation models using Multisim's Model Makers or code modeling.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Introduction to Component Editing

You can edit components in Multisim using the following methods:

- **Component Wizard**—Use to create and edit new components. Refer to the *Creating Components with the Component Wizard* section for more information.
- **Component Properties** dialog box—Use to edit existing components and is accessed from the **Database Manager** dialog box. Refer to the *Editing Components* section for more information.

You can modify any component stored in the Multisim component database. For example, an existing component might now be available in a new package (originally pin-through hole, now surface mount). You can easily copy the component information and change only the package details to create this new component. You can also create your own component and place it into the database or load a component from another source.

You cannot edit the **Master Database**. However, you can copy components to the **Corporate** or **User** database and then modify them as desired.

Tip Where possible, we recommend that you modify an existing, similar component, rather than create one.

Each component in the component database is identified by the following types of information, each of which is in a specific tab in the **Component Properties** dialog box:

- General information (such as name, manufacturer, date and author)—refer to the *Editing a Component's General Properties* section for more information.
- Symbol (pictorial representation of the component for schematic capture)—Refer to the *Editing a Component's Symbol* section for more information.
- Model (information used to represent the actual operation/behavior of the component during simulation)—Necessary only for a component that will be simulated. Refer to the *Editing a Component's Model* section for more information.
- Pin model (information used to represent the behavior of the pin during simulation). Refer to the *Editing a Component Pin Model* section for more information.
- Footprint (the package that Multisim uses when exporting a schematic containing this component to a PCB Layout package such as Ultiboard). Refer to the *Editing a Component's Footprint* section for more information.
- Electronic parameters of the component. Refer to the *Editing a Component's Electronic Parameters* section for more information.
- User fields (if used to further define the components). Refer to the *Editing User Fields* section for more information.

Components can be organized in the **Database Manager**, where you can:

- add and remove component families from the **User** or **Corporate** databases.
- modify user field titles for any database.
- add and change family icons.

Note If you modify any information about a component in the **Master Database**, you must store the modified information in the **User Database** or **Corporate Database** database.

Caution If you modify information about any component in the **Corporate Database** or **User Database**, you are prompted for a new name for the component. If you do not give a new name, Multisim saves the changes to the original location, so the original **User Database** or **Corporate Database** component information is overwritten.

Tip You can also edit components that you have already placed on the workspace. This can be done using the component's properties dialog box (double-click on the component to display) and changing parameters in the various tabs. Changes made to placed components in this manner will not be reflected in the database. If you place the same component from the database, it will contain the parameters that it had *before* it was edited. Edits to individual components can also be made via the **Spreadsheet View**.

Creating Components with the Component Wizard

Multisim includes a **Component Wizard** that quickly steps you through the process of creating a component for use in schematic capture, as well as simulation or layout, or both.

Creating an Analog Component

Analog parts such as diodes and transistors can be created following the procedure in this section.

You can also create resistors, inductors and capacitors. However, resistors, inductors and capacitors created using this procedure will only contain basic simulation model information. Those that are placed from the master database have additional temperature-related SPICE simulation parameters. Refer to the *Placing Resistors, Inductors or Capacitors* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Complete the following steps to create an analog component:

1. Click the **Create Component** button in the **Main** toolbar.

   Or

   Choose **Tools»Component Wizard**. Step 1 of the **Component Wizard** appears.

2. Complete the following:
   - **Component Name**—The value of the component. Examples include 10 ohms, 2N2221, 2uF. This field can only contain letters, numbers, and the following characters: `-+!@#$%^&()[]{}:`
   - **Author Name**—Completed by system; change if desired.
   - **Function**—A brief description of the component. This is useful because you can search the function field when looking for a specific type of component to place.
3. Select **Analog** from the **Component Type** drop-down list.

4. Select one of:
   - **I will use this component for both simulation and layout (model and footprint)**—To use the component for both simulation and PCB layout.
   - **Simulation only (model)**—To use the component for simulation only.
   - **Layout only (footprint)**—To use the component for PCB layout only.

Note The number of steps you must complete depends on the selections you make in step 1. If you choose to use the component for both simulation and layout, the wizard includes eight steps. If you choose to use the component for simulation only, the wizard includes seven steps; for layout only it includes six steps.

5. Click **Next**. The next step of the wizard appears, where you enter the component's footprint information (if applicable).
6. Click **Select a Footprint**. The **Select a Footprint** dialog box appears, where you select a footprint from one of the databases.
   - Select the desired **Database Name**, scroll down the list at the bottom of the dialog box and click the desired **Footprint**, for example, TO-39.

   Refer to the *Select a Footprint dialog box* section for more information on this dialog box.

Note If you do not wish to assign a footprint at this time, select the database where the component will be stored and click on the **Add** button. The **Add a Footprint** dialog box appears. Type in a name in the **Footprint** field, for example, placeholder. Enter "Generic" in the manufacturer field and click **OK**. Refer to the *Add a Footprint dialog box* section for more information. After you have named the footprint here, you can create it later in Ultiboard. Be sure to place the name in the correct field.

7. Click **Select**. You are returned to the **Component Wizard**. The **Footprint Manufacturer** and **Footprint Type** fields have been populated based on the selected footprint.

Note You can also enter values directly into the **Footprint Manufacturer** and **Footprint Type** fields instead of using the **Select a Footprint** dialog box. If you enter a value in either of these fields that does not exist in the **Master** or **User Database**, you will be prompted to confirm that you wish to add the information to a new footprint. If you select **Yes**, you will be presented with the **Add a Footprint** dialog box. Refer to the *Add a Footprint dialog box* section for more information.

8. Specify the **Number of Pins** for the component (three, in this example).
9. Select **Single Section Component** or **Multi-Section Component** as required. If you select **Multi-Section Component**, the **Number of Sections** and **Section Details** elements are added to the dialog. Complete these as required. You can create a multi-section component with up to 96 sections.
10. Click **Next** to display the next step of the wizard. This is where you enter symbol information for the component. The symbol you assign to this component appears when it is placed on the workspace.
    - In the **Symbol Set** area, select **ANSI** or **DIN**.
    - Click the **Copy from DB** button to browse the component databases if you wish to use an existing component symbol. Refer to the *Copying a Component's Symbol* section for more information.
    - Optionally, to modify the symbol, click **Edit** to launch the **Symbol Editor**. Refer to the *Creating and Editing a Component's Symbol with the Symbol Editor* section for more information.
    - Click **Copy To**, to use the same symbol for both the DIN and ANSI **Symbol Set**. You can also use this button to copy the symbol of a multi-section component to another section in that component. Refer to the *Copying a Multi-Section Component's Symbol* section for more information.
11. Click **Next** to display the next step of the **Component Wizard**. Enter the component's pin parameters as described below:
    - **Add Hidden Pin** button—Click to add hidden pins: **Power**; **Ground**; **Common**. A hidden pin is part of the model and/or footprint that does not show in the schematic.
    - **Section** column—Click in a field in this column and select the desired section for this pin. This is for multi-section components.
    - **Type** column—Click a field in this column and select the pin type from the drop-down list that appears. Choices are: **Passive**, **GND** (ground), **Bidirectional**, **Input**, **NC** (no connection), **Output** and **PWR** (power). These pin types affect the ERC report and the pin drivers/receivers for digital components.
    - **ERC Status** column—Click in a field in this column and select whether to include or exclude the pin from Electrical Rules Checking. Refer to the *Electrical Rules Checking* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.

12. To accept the symbol information displayed, click **Next**. This step is where you map symbol and footprint pin information. This mapping is needed for exporting to a layout package. A symbol pin is the name of the pin in the symbol, for example, VCC. The footprint pin is the number or name of that pin on the footprint in PCB layout. The symbol and footprint pins must accurately map in order to transfer correctly to PCB layout—refer to the *Pins* section for a more detailed explanation of symbol and footprint pins.
    - For each symbol pin, enter a corresponding footprint pin. The mapping information you enter here will be displayed in the **Footprint** tab of the **Component Properties** dialog box.
    - For advanced pin mapping options, click **Map Pins**. The **Advanced Pin Mapping** dialog box displays. Refer to the *Advanced Pin Mapping Dialog* section for more information.
13. Click **Next** to display the **Select Simulation Model** step. (If the component does not require simulation, this step does not appear.) The buttons in this step work as described below.

Note If you are creating a basic resistor, inductor or capacitor, skip this step.

- **Select from DB**—Displays the **Select Model Data** dialog box, where you copy model data from an existing component. Refer to the *Copying the Model of One Component to Another* section for more information.
- **Model Maker**—Displays the **Select Model Maker** dialog box, where you can select model makers that automatically generate simulation models based on datasheet values. Refer to the *Creating a Component Model Using the Model Makers* section for more information.
- **Load from File**—Displays a standard file browser where you navigate to, and select, the desired model file. Refer to the *Loading an Existing Model* section for more information.
- **Copy to**—Displays the **Select Target** dialog box. Use to copy model information from a selected section of a multi-section component to the target sections that you select in the **Select Target** dialog box. Refer to the *Copying a Multi-Section Component's Symbol* section for more information. This would typically be used after using the **Select from DB** button to copy model data from another component. This model data will have only been copied to the selected section of the component. By using **Copy to**, you can copy the information into the remaining sections of the component.

14. Click **Next** to display the next step, where you set mapping information between the symbol and the simulation model. (This is for parts with simulation models only.)
    - For each symbol pin enter a corresponding number to connect to its respective nodes in the model. The mapping information you enter here will be displayed in the **Model** tab of the **Component Properties** dialog box.

    If you are creating a basic resistor, inductor or capacitor, this dialog includes the following:
    - **SPICE Model Type** drop-down list—Select one of: Resistor(r); Capacitor(c); Inductor(l). The **Value** fields changes to reflect your selection.
    - **Value** fields—Enter the desired value, for example 100 uF.

Note A basic resistor, inductor or capacitor, created using the **Component Wizard** will only contain basic simulation model information. Those that are placed from the master database have additional temperature-related SPICE simulation parameters. Refer to the *Placing Resistors, Inductors or Capacitors* section of Chapter 2, *Schematic Capture - Basics*, for more information.

15. Click **Next**. The dialog box that appears lets you indicate where you would like the component to be saved. If there is no family in the group that you want to save the component, you can add a new family by clicking on the **Add Family** button. Refer to the *Managing Families* section of Chapter 5, *Components*, for more information.
16. Navigate to the family where you want to save the component and click **Finish**. The component is saved in the selected family.

Creating a Digital Component

Complete the following steps to create a digital component:

1. Click the **Create Component** button in the **Main** toolbar.

   Or

   Choose **Tools»Component Wizard**. Step 1 of the **Component Wizard** appears.
2. Complete the following as desired:
   - **Component Name**—The value of the component, for example, 74ALS00M. This field can only contain letters, numbers, and the following characters: `-+!@#$%^&()[]{}:`
   - **Author Name**—Completed by system; change if desired.

- **Function**—A brief description of the component.

3. Select digital from the **Component Type** drop-down list. The **Component Technology** drop-down list appears—Select the desired techology, for example, 74ALS.
4. Select one of:
   - **I will use this component for both simulation and layout (model and footprint)**—To use the component for both simulation and PCB layout.
   - **Simulation only (model)**—To use the component for simulation only.
   - **Layout only (footprint)**—To use the component for PCB layout only.

Note The number of steps you must complete depends on the selections you make in step 1. If you choose to use the component for both simulation and layout, the wizard includes eight steps. If you choose to use the component for simulation only, the wizard includes seven steps; for layout only it includes six steps.

5. Click **Next**. The next step of the wizard appears, where you enter the component's footprint information.
6. Click **Select a Footprint**. The **Select a Footprint** dialog box appears, where you select a footprint from one of the databases.
   - Select the desired **Database Name**, scroll down the list at the bottom of the dialog box and click the desired **Footprint**, for example, M14A.

   Refer to the *Select a Footprint dialog box* section for more information about this dialog box.

Note If you do not wish to assign a footprint at this time, select the database where the component will be stored and click on the **Add** button. The **Add a Footprint** dialog box appears. Type in a name in the **Footprint** field, for example, placeholder. Enter "Generic" in the manufacturer field and click **OK**. Refer to the *Add a Footprint dialog box* section for more information.

7. Click **Select**. You are returned to the **Component Wizard**. The **Footprint Manufacturer** and **Footprint Type** fields have been populated based on the selected footprint.

Note You can also enter values directly into the **Footprint Manufacturer** and **Footprint Type** fields instead of using the **Select a Footprint** dialog box. If you enter a value in either

of these fields that do not exist in the **Master** or **User Database**, you will be prompted to confirm that you wish to add the information to a new footprint. If you select **Yes**, you will be presented with the **Add a Footprint** dialog box. Refer to the *Add a Footprint dialog box* section for more information.

8. Select **Single Section Component** or **Multi-Section Component** as desired. If you select **Multi-Section Component**, the **Number of Sections** and **Section Details** elements are added to the dialog. Complete these as desired.
9. Click **Next** to display the next step of the wizard. This is where you enter symbol information for the component. The symbol you assign to this component appears when it is placed on the workspace.
   - In the **Symbol Set** area, select **ANSI** or **DIN**.
   - Optionally, to modify the symbol, click **Edit** to launch the **Symbol Editor**. Refer to the *Creating and Editing a Component's Symbol with the Symbol Editor* section for more information.
   - Click the **Copy from DB** button to browse the component databases if you wish to use an existing component symbol. Refer to the *Copying a Component's Symbol* section for more information.
   - Click **Copy To**, to use the same symbol for both the DIN and ANSI **Symbol Set**. You can also use this button to copy the symbol of a multi-section component to another section in that component. Refer to the *Copying a Multi-Section Component's Symbol* section for more information.
   - **Hidden Ground Pins**—Depending on the **Component Technology** selected in step 1, this section will have different options available (for example, GND). Select the desired radio button.
   - **Hidden Power Pins**—Depending on the **Component Technology** selected in step 1, this section will have different options available (for example, VCC). Select the desired checkboxes.
10. Click **Next** to display the next step of the **Component Wizard**. Enter the component's pin parameters as described below:
    - **Add Hidden Pin** button—click to add hidden pins: **Power**; **Ground**; **Common**, **Unconnected**.
    - **Section** column—Click in a field in this column and select the desired section for this pin. This is for multi-section components.

- **Type** column—Click a field in this column and select the pin type from the drop-down list that appears. Choices are: **Open Collector**, **3-State**, **GND** (ground), **Bi-Directional**, **Active Drive**, **Schmitt Trigger**, **Input**, **NC** (no connection), and **VCC**.
- **ERC Status** column—Click in a field in this column and select whether to **Include** or **Exclude** the pin from **Electrical Rules Checking**. Refer to the *Electrical Rules Checking* section for more information.

11. To accept the symbol information displayed, click **Next**. This is where you map symbol and footprint pin information. This mapping is needed for exporting to a layout package. A symbol pin is the name of the pin in the symbol, for example, VCC. The footprint pin is the number or name of that pin on the footprint in PCB layout. The symbol and footprint pins must accurately map in order to transfer correctly to PCB layout—refer to the *Pins* section for a more detailed explanation of symbol and footprint pins.
    - For each symbol pin, enter a corresponding footprint pin. The mapping information you enter here will be displayed in the **Footprint** tab of the **Component Properties** dialog box.
    - For advanced pin mapping options, click **Map Pins**. The **Advanced Pin Mapping** dialog box displays. Refer to the *Advanced Pin Mapping Dialog* section for more information.
12. Click **Next** to display the **Select Simulation Model** step. (If the component does not require simulation, this step does not appear). The buttons in this step work as described below.
    - **Select from DB**—Displays the **Select Model Data** dialog box, where you copy model data from an existing component. Refer to the *Copying the Model of One Component to Another* section for more information.
    - **Load from File**—Displays a standard file browser where you navigate to, and select, the desired model file. Refer to the *Loading an Existing Model* section for more information.
    - **Copy to**—Displays the **Select Target** dialog box. Use to copy model information from a selected section of a multi-section component to the target sections that you select in the **Select Target** dialog box. Refer to the *Copying a Multi-Section Component's Symbol* section for more information. This would typically be used after using the **Select from DB** button to copy model data from another component. This model data will have only been copied to the selected section of the component. By using **Copy to**, you can copy the information into the remaining sections of the component.

13. Click **Next** to display the next step, where you set mapping information between the symbol and the simulation model.
    - For each symbol pin enter a corresponding number to connect to its respective nodes in the model. The mapping information you enter here will be displayed in the **Model** tab of the **Component Properties** dialog box.
14. Click **Next**. The dialog box that appears lets you indicate where you would like the component to be saved. If there is no family in the group that you want to save the component, you can add a new family by clicking on the **Add Family** button. Refer to the *Managing Families* section of Chapter 5, *Components*, for more information.
15. Navigate to the family where you want to save the component and click **Finish**. The component is saved in the selected family.

Creating a VHDL Component

The **Component Wizard** creates VHDL parts in a similar manner to analog and digital components. For simplification, this section details making a VHDL component for simulation only.

Complete the following steps to create a VHDL component:

1. Click the **Create Component** button in the **Main** toolbar.

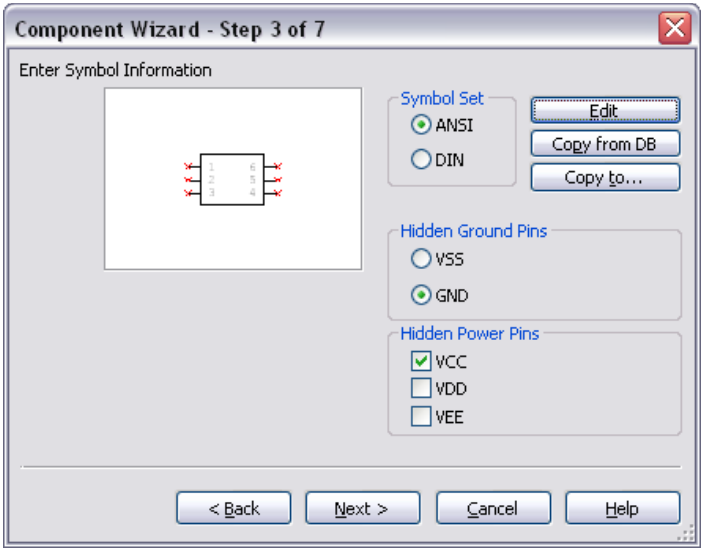
   Or

   Choose **Tools»Component Wizard**. Step 1 of the **Component Wizard** appears.
2. Complete the following:
   - **Component Name**—The value of the component. This field can only contain letters, numbers, and the following characters: `-+!@#$%^&()[]{}:`
   - **Author Name**—Completed by system; change if desired.
   - **Function**—A brief description of the component.
3. Select VHDL from the **Component Type** drop-down list. The **Component Technology** drop-down list appears with VHDL selected.
4. For this example, select **Simulation only (model)** and click **Next**.
5. Set the number of pins to be equal to the number of ports you have in the VHDL component. Don't count power or ground, only count the ports in the entity declaration.

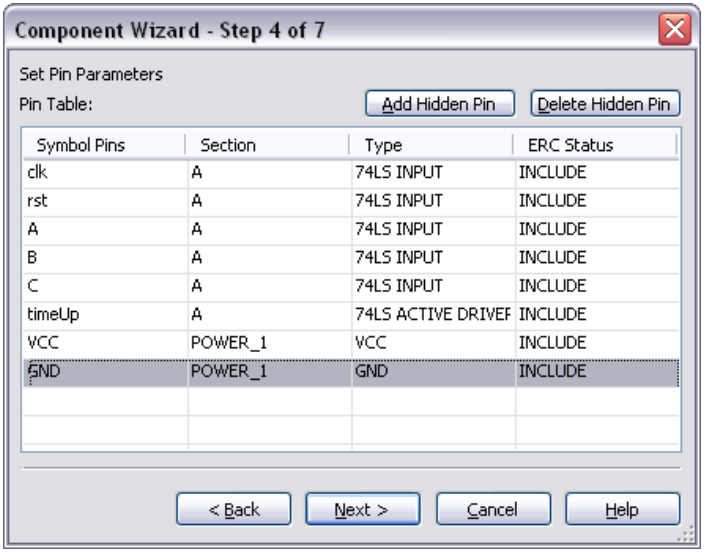
Note When you are creating your entity declaration in the VHDL source code, you must list inputs first, then bi-directional ports, and finally outputs. If you are using vectors than assign one pin to each bit in the vector. This section uses the following six-port example:

```
entity QUIZSHOW is
    port (
        clock: in std_ulogic;
        reset: in std_ulogic;
        contestantA: in std_ulogic;
        contestantB: in std_ulogic;
        contestantC: in std_ulogic;
        time_up: out std_ulogic
    );
end QUIZSHOW;
```

6. Click **Next** and change the symbol using the **Edit** or **Copy from DB** button. Refer to the *Creating and Editing a Component's Symbol with the Symbol Editor* section and the *Copying a Component's Symbol* section for more information. Be sure to select the desired **Hidden Ground Pins** and **Hidden Power Pins**.



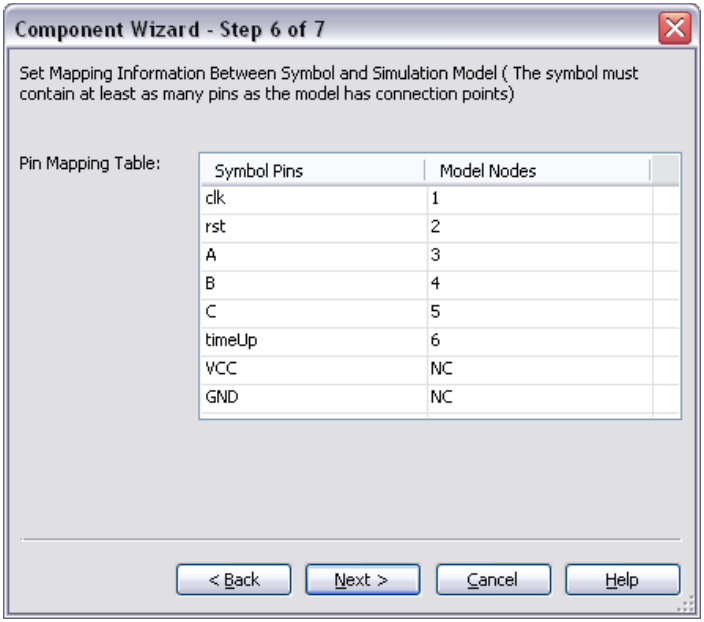
7. Click **Next** and select the pin drivers to use with the component. This is done in the **Type** column as shown below. Be sure use the **Add Hidden Pin** button to add the power and ground pins, using the same names that you selected in the previous step under **Hidden Ground Pins** and **Hidden Power Pins**.


Component Wizard - Step 4 of 7
Set Pin Parameters
Pin Table:
Add Hidden Pin
Delete Hidden Pin
Symbol Pins | Section | Type | ERC Status
clk | A | 74LS INPUT | INCLUDE
rst | A | 74LS INPUT | INCLUDE
A | A | 74LS INPUT | INCLUDE
B | A | 74LS INPUT | INCLUDE
C | A | 74LS INPUT | INCLUDE
timeUp | A | 74LS ACTIVE DRIVEF | INCLUDE
VCC | POWER_1 | VCC | INCLUDE
GND | POWER_1 | GND | INCLUDE
< Back
Next >
Cancel
Help

8. Click **Next** to display the next dialog where you will load the part's model. Enter the desired **Model Name**, click **Load from File**, and navigate to the VHDL file you wish to use. This will have a `.vx` extension. Make sure that you load the source file from MultiVHDL. See below.

Be sure that the **Model Name** matches the model name in the **Model Data** area.

9. Click **Next**. In this step you will relate the symbol pins to the model nodes. For VHDL, the order of the model pins is the order that they are listed in the port declarations of the entity. Earlier in this procedure, you see that the order of the model pins used in this example is: 1 - clock; 2 - reset; 3 - contestantA; 4 - contestantB; 5 - contestantC; 6 - time_up. The symbol names do not need to be identical, but the order here is critical. Don't forget to change the VCC and GND **Symbol Pin**'s **Model Nodes** to VCC and GND.



10. Click **Next** and save the component to the database in the same manner used for analog devices.
11. When using the component, be sure to place the appropriate power and ground components in the schematic.

Note Verilog HDL parts can be created in a manner similar to that of VHDL parts. You will need to have the MultiHDL Verilog application that generates the Verilog source code referenced in the simulation model selection step.

Using a Symbol File Created in the Symbol Editor

This section demonstrates how to use a symbol that you create in the **Symbol Editor** when creating a new component using the **Component Wizard**.

Complete the following steps to create a symbol file:

1. Select **Tools»Symbol Editor**.
2. In the symbol editor, create the desired symbol. Refer to the *Creating and Editing a Component's Symbol with the Symbol Editor* section for more information.
3. Select **File»Save As** and save the symbol into an easily accessible location such as your Desktop.

Complete the following steps to create a new component with the symbol saved in the above step:

1. Select **Tools»Component Wizard**.
2. Complete step 1 of the **Component Wizard** and click **Next**.
3. In step 2 of the **Component Wizard**, enter the desired footprint and number of pins.
4. In step 3 of the component wizard, click the **Edit** button. The **Symbol Editor** appears.
5. Select **File»Open** and open the symbol file saved earlier.
6. Select **File»Exit**. You are prompted to save the symbol. Click **Yes** to save the new symbol for the new component.

Note If you make any changes to the symbol , and select **File»Save**, the original symbol will be overwritten with the changes. If you only wish to make changes for the specific component you are creating, be sure to select **File»Exit**.

7. Continue with the remaining steps in the component wizard. Refer to the *Creating Components with the Component Wizard* section for more information.

Editing Components

Complete the following steps to edit an existing component:

1. Click the **Database Manager** button on the **Main** toolbar.

   Or

   Choose **Tools»Database»Database Manager**.

   The **Database Manager** dialog box appears.
2. Select the **Components**.
3. In the **Component List**, select the component you want to edit.
4. Click **Edit**. The **Component Properties** dialog box appears, consisting of multiple tabs.
5. Edit component information in the various tabs as described in the sections referenced below:
   - *Editing a Component's General Properties*.
   - *Editing a Component's Symbol*.
   - *Editing a Component's Model*.
   - *Editing a Component Pin Model*.

- *Editing a Component's Footprint*.
- *Editing a Component's Electronic Parameters*.
- *Editing User Fields*.

6. Click **OK** from any of the tabs to save the edited component. The **Select Destination Family Name** dialog box appears.
7. In the **Family Tree** area:
   - Choose the database where you want to store the edited component. Remember, you cannot change information in, or add to, the **Master Database**, so you must make your changes in either the **Corporate Database** or **User Database**.
   - Choose the **Group** in which you want the edited component stored and then click the **Family** name. The **OK** button becomes active.

Note If you wish to add a family to the selected database, click the **Add Family** button. The **New Family Name** dialog box appears where you enter the desired **Family** information. Refer to the *Managing Families* section of Chapter 5, *Components*, for more information.

8. Click **OK** to close the **Select Destination Family Name** dialog box.

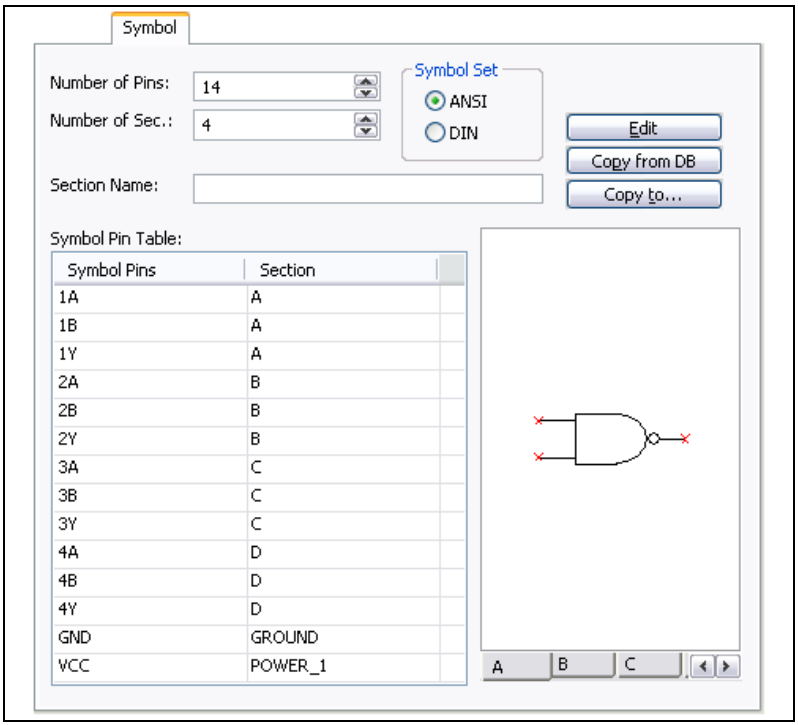
Editing a Component's General Properties

The **General** tab of the **Component Properties** dialog box allows you to modify the component's name and change the functional description. The date and author are drawn from system information and cannot be changed.

Editing a Component's Symbol

The **Symbol** tab of the **Component Properties** dialog box allows you to:

- edit a component's symbol
- give a component the same symbol as another component
- create a symbol for a component.



Set the following as desired:

- **Number of Pins**—The number of pins in the component.
- **Number of Sec.**—The number of sections in the component.
- **Section Name**—This is optional.
- **Symbol Pin Table**—This is where you map the **Symbol Pins** to the desired **Section**. Pins like VCC and GND are common to all gates in the IC.
- **Edit** button—Use to edit the component's symbol.
- **Copy from DB** button—Use to copy a symbol from another component.
- **Copy to** button—Use to copy a symbol to another section of the component.

Copying a Component's Symbol

Complete the following steps to copy a symbol from another component:

1. From the **Symbol** tab of the **Component Properties** dialog box, click **Copy from DB**. The **Select a Symbol** dialog box appears.
2. Navigate through the **Database**, **Group**, **Family** and **Component** lists to identify the component whose symbol you want to copy and click **OK**. (Click **Close** to cancel.) Multisim returns to the **Component Properties** dialog box, where the symbol associated with the selected component appears.
3. If you want to change the symbol set to be associated with this component, select **ANSI** or **DIN** from the **Symbol Set** box. The appropriate symbol appears in the upper part of the dialog box.
4. To confirm the association of this symbol with your component, click **OK**.

Copying a Multi-Section Component's Symbol

Complete the following steps to copy the symbol of a multi-section component to another section in that component:

1. Click **Copy to** in the **Symbol** tab of the **Component Properties** dialog box.

   Or

   Click **Copy to** in the **Create Component Wizard - Step 3 of 7** dialog box.

   The **Select Target** dialog box displays.
2. Select the sections to where you would like to copy the symbol and click **OK**.

Note The **Select Target** dialog box is also accessed from the **Select Simulation Model** dialog box of the **Create Component Wizard**. In this case you are copying model information from a selected section of a multi-section component to the target sections that you select in the **Select Target** dialog box.

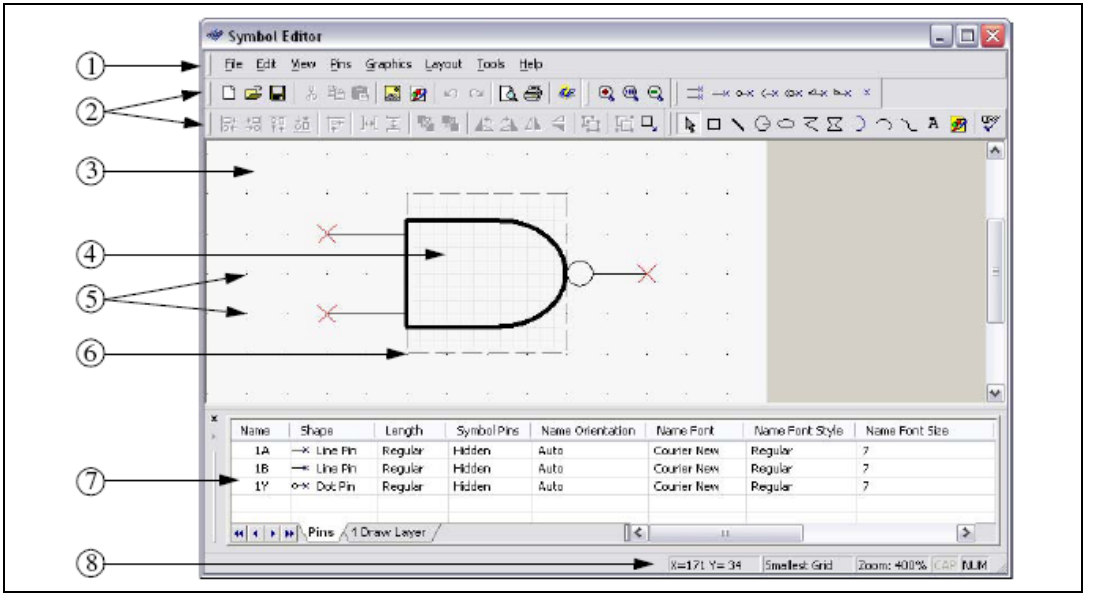
Creating and Editing a Component's Symbol with the Symbol Editor

The **Symbol Editor** is a specialized graphics editor that allows you to create or modify a component's symbol. The **Symbol Editor** functions in the "normal" mode when editing component symbols in the database, or creating a new symbol. To edit the symbol for a single component in your active circuit, use the **In-Place Edit Mode**. Refer to the *In-Place Edit Mode* section for more information.

Complete the following steps to edit a component symbol:

1. Double-click on the desired component and click **Edit Component in DB** to display the **Component Properties** dialog box.
2. In the **Symbol** tab of the **Component Properties** dialog box, in the **Symbol Set** box, be sure you have selected the desired symbol set (**ANSI** or **DIN**).
3. Click **Edit**. The **Symbol Editor** appears, displaying the selected symbol for you to edit.

The **Symbol Editor** looks like this:


Symbol Editor
File Edit View Pins Graphics Layout Tools Help
Name Shape Length Symbol Pins Name Orientation Name Font Name Font Style Name Font Size
1A Line Pin Regular Hidden Auto Courier New Regular 7
1B Line Pin Regular Hidden Auto Courier New Regular 7
1Y Dot Pin Regular Hidden Auto Courier New Regular 7
Pins
1 Draw Layer
X=171 Y=34
Smallest Grid
Zoom: 400%
CAP
NUM

1	Menu Bar	3	Workspace	5	Pin Grid	7	Spreadsheet View
2	Toolbars	4	Draw Grid	6	Boundary Box	8	Status Bar

The **Symbol Editor** dialog box consists of:

- the **Menu Bar** (1), which contains the menus with their associated commands.
- the **Toolbars** (2), which give quick access to the most commonly-used tools.
- the **Workspace** (3), which is where you build or modify your symbols. The **Pin Grid** (5) aids in the placement of pins. Its size cannot be adjusted. The **Draw Grid** (4) aids in the placement of graphic elements inside the **Boundary Box** (6).

- the **Spreadsheet View** (7), which is where you find and edit various graphical primitives and pin parameters.
- the **Status Bar** (8), which gives information on the commands and actions associated with the position of the cursor.

The next sections describe the menus, toolbars and **Symbol Editor** functionality in more detail.

Symbol Editor Spreadsheet View

The **Spreadsheet View** is where you find and edit various symbol parameters. When you select an item on the workspace, it is highlighted in the spreadsheet, and vice versa. If you make a change to an item in the spreadsheet, it is reflected on the symbol in the workspace.

Pins Tab

The **Pins** tab of the **Symbol Editor Spreadsheet View** contains the following columns:

- **Name**—The pin's name. You can perform full or partial negation on pin names by using the ^ symbol, as shown in the examples below:
  - ^MyPin negates all of the text. In this example, the pin name displays with a negation bar above all of MyPin.
  - ^MyPin^ also negates all of the text.
  - My^Pin^ negates Pin only—a negation bar appears above Pin.
- **Shape**—Select a row and click in this field to display a list of pin types. Choices are: **Line Pin**; **Dot Pin**; **Clock Pin**; **Dot-Clock Pin**; **Input Wedge Pin**; **Output Wedge Pin**; **Zero-Length Pin**. Select the desired pin type. The symbol changes to reflect your selection. Refer to the *Pins* section for more information about pins.
- **Length**—Select a row and click in this field to display a list of pin lengths. Choices are: **Short** (1 grid); **Regular** (2 grids); **Long** (3 grids); **Extra Long** (4 grids). If **Zero-Length Pin** is selected in the **Shape** field, the list of pin lengths is not available in the **Length** field.
- **Symbol Pins**—Select a row and click in this field and select either **Hidden** or **Visible** to hide or show the pin name on the symbol.
- **Name Orientation**—Select a row and click in this field to set the orientation of the pin's name on the symbol. Choices are: **Auto**; **Vertical**; **Horizontal**.
- **Name Font**—The font used for the pin's name.

- **Name Font Style**—The style of font used for the pin's name (for example, **Bold**).
- **Name Font Size**—The size of the font used for the pin's name.
- **Footprint Pins**—Select a row and click in this field and select either **Hidden** or **Visible** to hide or show the footprint pin on the symbol.
- **Number Orientation**—Select a row and click in this field to set the orientation of the pin number on the symbol. Choices are: **Auto**; **Vertical**; **Horizontal**.
- **Number Font**—The font used for the pin's number.
- **Number Font Style**—The style of font used for the pin's number.
- **Number Font Size**—The size of the font used for the pin's number.

Draw Layer Tab

The **Draw Layer** tab of the **Symbol Editor Spreadsheet View** contains the following columns:

- **Name**—The type of graphic element, for example, arc.
- **Pen Type**—The appearance of lines for graphics elements, excluding placed text. Select a row and click in this field to display a list of pen types. Choices are: **Solid**; **Dash**; **Dot**; **Dash-Dot**; **Dash-Dot-Dot**; **Invisible**; **Solid Inside Frame**.
- **Pen Width**—The width of the lines in graphic elements, excluding placed text. Select a row and click in this field to display a list of pen types. Choices are: **Not Scalable**; **One Pixel**; **Two Pixels**; **Three Pixels**; **Four Pixels**; **Five Pixels**.
- **Pen Color**—The color of lines for graphic elements, including placed text. Select a row and click in this field to display a color palette.
- **Brush Type**—The style of the fill in elements such as polygons, that have a fill. Select a row and click in this field to display a list of fill types. Choices are: **Solid**; **Invisible**; **Horizontal**; **Vertical**; **Diagonal Downward**; **Diagonal Upward**; **Cross**; **Diagonal Cross**.
- **Brush Color**—The color of the fill in elements such as polygons, that have a fill. Select a row and click in this field to display a color palette.
- **Font**—The font name, active for placed text elements only. Select a row and click in this field to display a list of fonts.
- **Font Style**—The font style, active for placed text elements only. Select a row and click in this field to display a list. Choices are: **Regular**; **Italic**; **Bold**; **Bold Italic**.

- **Font Size**—The font size, active for placed text elements only. Select a row and click in this field to display a list of sizes.

Note If objects are grouped using **Edit»Group**, their distinct names and properties will no longer appear. The name for any grouped object will appear as “Group”.

Working with the Symbol Editor

To use the **Symbol Editor** to create working symbols you need to be familiar with the elements required to make up a symbol in Multisim. The key elements needed for a symbol are:

- shape
- pins

These are described in more detail below.

Shape

A symbol requires a shape to allow users to recognize its general function.

Use the drawing capabilities of Multisim’s **Symbol Editor** to construct a shape that makes logical sense for the component you are creating or modifying. The simplest way to do this is to edit the shape of an existing component. Once this is done, you must add pins.

Pins

There are three main parts to a pin. These are the:

- **Name**—This is the name of the pin and is usually the same as that given in the data sheet. For example, most digital parts have pins named “VCC” and “GND”. These names are used to identify the actual name of the pin in the symbol and must be unique. It is recommended that you use concise names as they display best on the workspace and on printed output.

  There are two methods to handle components that have more than one pin with the same name:

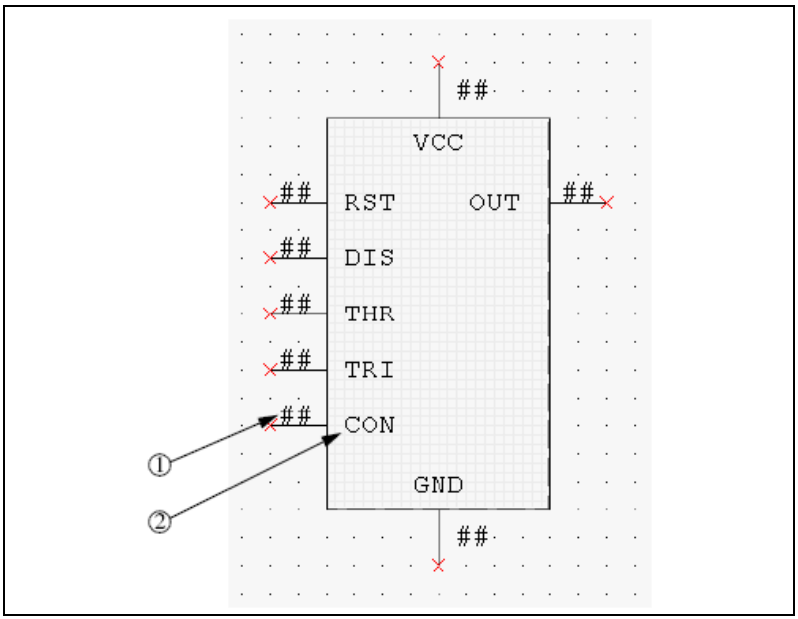
  – Have one pin on the symbol that represents the duplicated pin (for example, “GND”) and match the footprint pins later to the one on the symbol pin.

  – Add multiple pins that have the same basic name but append a digit to represent each occurence. For example, you could represent multiple ground connections as GND1, GND2, GND3, etc.

- **Footprint Pin Name**—This is the pin name on the footprint for the device. This naming will match the naming used in PCB layout. Within the **Symbol Editor** you only choose whether to display the footprint pin names by default, or not, and their orientation on the workspace.
- **Shape**—The shape of the pin denotes the type of pin. Multisim provides seven pin shapes that you can use: **Line Pin**; **Dot Pin**; **Clock Pin**; **Dot-Clock Pin**; **Input Wedge Pin**; **Output Wedge Pin**; **Zero-Length Pin**. Refer to the *Place Pin Toolbar* section for more information. The shapes are used for identification purposes and do not have any impact on the operation of the component.

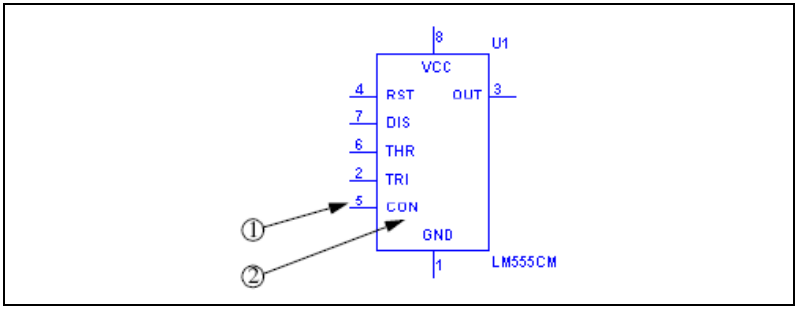
Note The symbol and footprint pins must accurately map in order to transfer correctly to PCB layout.

The figure below shows a 555 timer's symbol, as seen in the **Symbol Editor**. Both the footprint pins (1) and the symbol pins (2) appear.



1	Footprint Pin	2	Symbol Pin

The figure shows the same 555 timer's symbol, as seen in the Multisim workspace. The footprint pins (1) and the symbol pins (2) appear slightly differently.



1	Footprint Pin	2	Symbol Pin

The relationship of the Footprint Pin to the Symbol Pin is mapped in the **Footprint** tab of the **Component Properties** dialog box. Refer to the *Editing a Component's Footprint* section for more information.

Complete the following steps to add a pin to a symbol:

1. Click on the desired pin type in the **Place Pin Toolbar**.
2. Click on the workspace where you want the pin to appear.

Note Pins cannot be placed on top of each other or on corners of the boundary box. If you attempt to do so, the cursor switches to a circle with a line through it, indicating that the action is prohibited.

Note You can edit a pin's properties in the **Pins** tab of the **Spreadsheet View**. Refer to the Pins Tab section for more information.

Default Pin Options Dialog Box

Note Refer to the *Pin Array Options Dialog Box* section for information about the **Pin Array Options** dialog box.

Complete the following steps to set up the default options for pins:

1. Choose **Pins»Default Pin Options** to display the **Default Pin Options** dialog box.
2. Set the options in the **Pin Name** box:
   - **Prefix** field—Appears before the pin name. A preview of the prefix appears in the **Name Preview** field.

- **Suffix** field—Appears after the pin name. A preview of the suffix appears in the **Name Preview** field.
- **Index** checkbox—Enables the **Start from** and **Increment by** fields.
- **Start from** field—The starting number for the pins.
- **Increment by** field—The size of the steps between pin numbers.

3. Set the options in the **Pin Properties** box.
   - **Shape** drop-down list—The shape of the pin.
   - **Length** drop-down list—The length of the pin.
   - **Name Orientation** drop-down list—The orientation of the pin name. If you do not wish to show the pin name, turn off the **Name Visible** checkbox.
   - **Number Orientation** drop-down list—The orientation of the pin number. If you do not wish to show the pin number, turn off the **Number Visible** checkbox.
4. Optionally, enable the **Continuous Pin Placement** checkbox. This allows you to place pins one after the other, based on the parameters entered in the above steps, until the <Esc> button on your keyboard is pressed.

Pin Array Options Dialog Box

Complete the following steps to place a pin array on a symbol:

1. Choose **Pins»Place Pin Array** to display the **Pin Array Options** dialog box.
2. Set the options in the **Pin Name** box:
   - **Prefix** field—Appears before the pin names. A preview of the prefix appears in the **Name Preview** field.
   - **Index** checkbox—Enables the **Start from** and **Increment by** fields.
   - **Start from** field—The starting number for the pins in the array.
   - **Increment by** field—The size of the steps between pin numbers in the array.
   - **Suffix** field—Appears after the pin names. A preview of the suffix appears in the **Name Preview** field.
3. Enter the quantity of the pins desired in the **Number of Pins in Array** field.

4. Enter the spacing of the pins (in grids) in the **Distance between Pins in Array** field.
5. Set the options in the **Pins Order** box:
   - **Clockwise** button—Pins are placed in ascending order clockwise.
   - **Counterclockwise** button—Pins are placed in ascending order counter-clockwise.
6. Set the options in the **Pin Properties** box:
   - **Shape** drop-down list—The shape of the pin.
   - **Length** drop-down list—The length of the pin.
   - **Name Orientation** drop-down list—The orientation of the pin name. If you do not wish to show the pin name, turn off the **Name Visible** checkbox.
   - **Number Orientation** drop-down list—The orientation of the pin number. If you do not wish to show the pin number, turn off the **Number Visible** checkbox.
7. Click **OK** and place the array in the desired location. You are not permitted to place the array if there is insufficient space.

Enter Text Dialog Box

This dialog box is accessed from either the **Symbol Editor**, or **Title Block Editor**. Refer to the *Title Block Editor* section of Chapter 3, *Schematic Capture - Advanced Functions*, for information on the **Title Block Editor**.

Complete the following steps to enter text on a symbol or title block:

1. Select **Graphics»Text** to display the **Enter Text** dialog box.
2. Type the desired text in the **Enter Text** field.
3. Change the formatting of the text as desired:
   - **Font** field—Select desired font from the list or type in.
   - **Font Style** field—Select desired style from the list.
   - **Size** field—Select desired size from the list or type in.
   - **Text Orientation** box—Select either horizontal or vertical orientation.
   - **Automatic** drop-down—Optionally, select a new color from the pop-up that appears when you click on the down-arrow.
4. Click **OK** and place the text in the desired location.

Caution If the boundary does not contain sufficient space to place the text, it will be lost when you click the mouse.

In-Place Edit Mode

To edit the symbol for a component placed on the workspace, right-click on the component and select **Edit Symbol** from the pop-up that appears. The **Symbol Editor** launches in the **In-Place Edit Mode**.

You can edit the symbol via the menus, toolbars and spreadsheet.

In the **In-Place Edit Mode**, you can change the graphical elements of a symbol (for example, the shape and placed text). You cannot add or remove pins as this would affect the footprint and model mapping.

Note The **In-Place Edit Mode** only changes the symbol for the selected component. It does not change the symbol for other components of the same value within the active circuit, or in the database.

Symbol Editor Menus

The **Symbol Editor** menus contain all the commands necessary to create and edit component symbols.

File Menu

The following selections are available under the **File** menu:

- **New**—Opens a new untitled document in the **Symbol Editor**. If you already have one open, it will close first, after prompting you to save any changes.
- **Open**—Opens an existing document in the **Symbol Editor**. If you already have one open, it will close first, after prompting you to save any changes.
- **Save**—Saves changes to the active document.
- **Save As**—Opens the standard Windows “Save As” dialog box where you can save the active document under a new or existing name.
- **Print Setup**—Opens the standard Windows “Print Setup” dialog box where you can enter the desired parameters for your printer.
- **Print Preview**—Opens the **Print Preview** dialog box, which shows the symbol in the active document with the best fit to the page. There are no other sizes available.
- **Print**—Opens the standard Windows “Print” dialog box where you can enter the desired printing properties and print the symbol. The symbol is printed with the best fit to the page. There are no other sizes available for printing the symbol.

- **Exit**—Closes the **Symbol Editor** and returns you to the main Multisim screen. Before exiting, you are prompted to save changes, if any, to the active document.

Edit Menu

The following selections are available under the **Edit** menu:

- **Undo**—Undoes the previous action.
- **Redo**—Redoes the previous "undo" action.
- **Cut**—Removes the selected element(s) from the workspace and places them on the clipboard.
- **Copy**—Places a copy of the selected element(s) on the clipboard. Note that pin names cannot be copied separately.
- **Paste**—Places a copy of the element(s) on the clipboard on the workspace at the cursor's location. If the clipboard contains pins and graphical elements, only the graphical elements will be pasted.
- **Delete**—Removes the selected element(s) from the workspace. They are *not* placed on the clipboard.
- **Copy As Picture**—Copies the symbol on the workspace to the clipboard as a metafile. For use with Winword.
- **Copy As Bitmap**—Copies the symbol on the workspace to the clipboard as a bitmap image. For use with MS PaintBrush.
- **Select All**—Selects all of the elements on the workspace.
- **Flip Horizontal**—Flips the selected element(s) horizontally. Does not apply to pins.
- **Flip Vertical**—Flips the selected element(s) vertically. Does not apply to pins.
- **Rotate 90 Clockwise**—Rotates the selected element(s) 90 degrees clockwise. Does not apply to pins.
- **Rotate 90 Counter CW**—Rotates the selected element(s) 90 degrees counter-clockwise. Does not apply to pins.
- **Snap To Grid**—Snaps the selected element(s) to the **Draw Grid** that is found within the symbol's boundary box. Does not apply to pins.
- **Group**—Places selected elements in one group. Does not apply to pins.
- **Ungroup**—Returns an element that was made using the **Group** command, back to its individual elements. Does not apply to pins.

- **Bring To Front**—Brings selected element(s) to the foreground on the workspace. Other element(s) appear behind them. Does not apply to pins.
- **Send To Back**—Sends selected element(s) to the background on the workspace. Other elements appear in front of them. Does not apply to pins.
- **Resize Boundary Box**—Places a cursor at the lower-right side of the boundary box. Drag it to the desired location to resize the boundary box. You cannot make the boundary box smaller than the elements that it contains or smaller than is required for the pins that are attached to it.

View Menu

The following selections are available under the **View** menu:

- **Toolbars**—Toggles the following toolbars on and off: **Standard Toolbar**; **Zoom Toolbar**; **Draw Tools**; **Pin Place Toolbar**; **Drawing Toolbar**.
- **Spreadsheet**—Toggles the spreadsheet view on and off. Refer to the *Symbol Editor Spreadsheet View* section for more information.
- **Status Bar**—Toggles the **Status** bar at the bottom of the screen on and off.
- **Show Pin Grid**—Toggles the **Pin Grid**, which displays outside of the **Boundary Box** on the workspace, on and off. For consistency throughout Multisim, the **Pin Grid** size cannot be changed.
- **Show Draw Grid**—Toggles the **Draw Grid**, which displays inside the **Boundary Box**, on and off.
- **Draw Grid Size**—Sets the size of the **Draw Grid**, which displays inside the **Boundary Box**. The choices are: **No Grid**; **Smallest Grid**; **Small Grid**; **Regular Grid**; **Large Grid**.
- **Zoom In**—Magnifies the element(s) in the workspace.
- **Zoom Out**—Reduces the viewing size of the element(s) in the workspace.
- **Zoom 100%**—Displays the items in the workspace at their normal viewing size. This is the size that they will be displayed at in Multisim. When the **Symbol Editor** first opens, the magnification is set to 200%.
- **Center By Mouse**—When viewing the workspace at high magnifications, you can use this command to center the image on the workspace. Select **Center By Mouse** and then click on the spot that you would like to be placed at the center of the workspace.
- **Redraw**—Redraws all elements in the workspace.

Pins Menu

The following selections are available under the **Pins** menu:

- **Select**—Use to select element(s) on the workspace by clicking and dragging the mouse.
- **Line Pin**—Places a line pin on the workspace.
- **Dot Pin**—Places a dot pin on the workspace.
- **Clock Pin**—Places a clock pin on the workspace.
- **Dot-Clock Pin**—Places a dot-clock pin on the workspace.
- **Input Wedge Pin**—Places an input wedge pin on the workspace.
- **Output Wedge Pin**—Places an output wedge pin on the workspace.
- **Zero-Length Pin**—Places a zero-length pin on the workspace.
- **Place Pin Array**—Displays the **Pin Array Options** dialog box where you enter parameters to place a pin array on the workspace. Refer to the *Pin Array Options Dialog Box* section for more information.
- **Default Pin Options**—Displays the **Default Pin Options** dialog box where you enter the default settings for the various pin types. Refer to the *Default Pin Options Dialog Box* section for more information.

Note Refer to the Pins section for information about pin types.

Graphics Menu

The following selections are available under the **Graphics** menu:

- **Text**—Displays the **Enter Text** dialog box where you enter and format text to be placed on the workspace. Refer to the *Enter Text Dialog Box* section for more information.
- **Line**—Draws a line on the workspace.
- **Multiline**—Draws a multiline on the workspace.
- **Half Ellipse Arc**—Places half of an ellipse on the workspace. Click once at the desired starting point and click again where you wish the diameter of the ellipse to end. As you move the cursor, a dotted outline of the ellipse appears. Click again to place the outer point of the arc at the desired location.
- **Segment Arc**—Places an arc on the workspace. Click once to place the center point of the arc, click again to place the outer diameter point. As you move the cursor, a dotted outline of the arc appears. Now move the cursor and click to place the arc's end point.

- **Bezier**—Places a bezier curve on the workspace. Click to place the start of the curve, then move the cursor on the workspace. As you move the cursor, a dotted outline of the bezier curve appears. Click to place the end point of the bezier, then move and click the cursor twice more to form the final shape of the bezier.
- **Rectangle**—Places a rectangle on the workspace.
- **Circle**—Places a circle on the workspace. Click to place the center of the circle, then move the cursor on the workspace. As you move the cursor, a dotted outline of the circle appears. When the circle is the desired size, click to place it on the workspace.
- **Ellipse**—Places an ellipse on the workspace. Click to place the center of the ellipse, then move the cursor on the workspace. As you move the cursor, a dotted outline of the ellipse appears. When the ellipse is the desired shape and size, click to place it on the workspace.
- **Polygon**—Places a polygon on the workspace. Click to place the polygon's corners and right-click to finish.
- **Bitmap**—Places a bitmap image on the workspace. Click to display a standard Windows "Open" dialog box, where you can select the desired bitmap.

Note Graphical primitives such as lines and rectangles can only be placed inside the boundary box. If you attempt to place them outside of the boundary box, the cursor will change to a circle with a line through it, indicating that the action is prohibited.

Layout Menu

To align two or more graphic elements, use one of the following selections from under the **Layout»Align** menu:

- **Left**—Left aligns the selected elements.
- **Horiz. Center**—Horizontally centers the selected elements.
- **Right**—Right aligns the selected elements.
- **Top**—Aligns the selected elements along their top edges.
- **Vert. Center**—Vertically centers the selected items.
- **Bottom**—Aligns the selected elements along their bottom edges.

Tools Menu

The following selection is available under the **Tool** menu:

- **Check Symbol**—Confirms that the symbol has no errors, such as duplicate pin names.

- **Customize**—Displays the **Customize** dialog box. Refer to the *Customizing the Interface* section of Chapter 1, *User Interface*, for more information.

Help Menu

The following selections are available under the **Help** menu:

- **Help Topics**—Displays the help file.
- **About Symbol Editor**—Displays a splash window with information about the **Symbol Editor**.

Pop-up Menus

Depending on where you right-click in the **Symbol Editor**, different pop-up menus appear.

Right-click from a menu to display a pop-up which allows you to toggle the following toolbars on and off: **Standard Toolbar**; **Zoom Toolbar**; **Draw Tools**; **Pin Place Toolbar**; **Drawing Toolbar**.

Right-click in the workspace to display a pop-up that contains: **Cut**; **Copy**; **Paste**; **Show Draw Grid**; **Draw Grid Size**; **Snap To Grid**; **Flip Horizontal**; **Flip Vertical**; **Rotate 90 Clockwise**; **Rotate 90 Counter CW**. Refer to the *Edit Menu* section for information about these commands.

Symbol Editor Toolbars

The toolbars give access to the most commonly-used tools.

Standard Toolbar

The buttons in the **Standard** toolbar found in the **Symbol Editor** are described below:

Button	Description
	New button. Opens a new untitled document in the **Symbol Editor**. If you already have one open, it will close first, after prompting you to save any changes.
	Open button. Opens an existing document in the **Symbol Editor**. If you already have one open, it will close first, after prompting you to save any changes.

Button	Description
	Save button. Saves changes to the active document.
	Cut button. Removes the selected element(s) from the workspace and places them on the clipboard.
	Copy button. Places a copy of the selected element(s) on the clipboard.
	Paste button. Places a copy of the element(s) on the clipboard on the workspace at the cursor's location.
	Copy As Picture button. Copies the symbol on the workspace as a metafile.
	Copy As Bitmap button. Copies the symbol on the workspace as a bitmap image.
	Undo button. Undoes the previous action.
	Redo button. Redoes the previous "undo" action.
	Print Preview button. Opens the **Print Preview** screen, which shows the symbol in the active document with the best fit to the page. There are no other magnifications available.
	Print button. Opens the standard Windows "Print" dialog box, where you can enter the desired printing properties and print the symbol. The symbol is printed with the best fit to the page. There are no other magnifications available for printing the symbol.
	About Symbol Editor button. Displays a splash screen with information about the **Symbol Editor**.

Zoom Toolbar

The buttons in the **Zoom** toolbar found in the **Symbol Editor** are described below.

Button	Description
	Zoom In button. Magnifies the elements on the workspace.
	Zoom 100% button. Displays the items in the workspace at their normal viewing size. This is the size that they will be displayed at in Multisim. When the **Symbol Editor** first opens, the magnification is set to 200%.
	Zoom Out button. Reduces the viewing size of the element(s) on the workspace.

Draw Tools Toolbar

The buttons in the **Draw Tools** toolbar found in the **Symbol Editor** are described below:

Button	Description
	Select button. Use to select element(s) on the workspace by clicking and dragging the mouse.
	Rectangle button. Places a rectangle on the workspace.
	Line button. Places a line on the workspace.
	Circle button. Places a circle on the workspace. Click to place the center of the circle, then move the cursor on the workspace. As you move the cursor, a dotted outline of the circle appears. When the circle is the desired shape and size, click to place it on the workspace.

Button	Description
	Ellipse button. Places an ellipse on the workspace. Click to place the center of the ellipse, then move the cursor on the workspace. As you move the cursor, a dotted outline of the ellipse appears. When the ellipse is the desired shape and size, click to place it on the workspace.
	Multiline button. Places a multiline on the workspace.
	Polygon button. Places a polygon on the workspace.
	Half Ellipse Arc button. Places a half ellipse arc on the workspace.
	Segment Arc button. Places an arc on the workspace. Click once to place the center point of the arc, click again to place the out diameter point. As you move the cursor, a dotted outline of the arc appears. Now move the cursor and click to place the arc's end point.
	Places a bezier curve on the workspace. Click to place the start of the curve, then move the cursor on the workspace. As you move the cursor, a dotted outline of the bezier curve appears. Click to place the end point of the bezier, then move and click the cursor twice more to form the final shape of the bezier.
	Text button. **Displays the Enter Text** dialog box, where you enter and format text to be placed on the workspace. Refer to the *Enter Text Dialog Box* section for more information.
	Bitmap button. Places a bitmap image on the workspace. Click to display a standard Windows "Open" dialog box, where you can select the desired bitmap.
	Check Symbol button. Confirms that the symbol has no errors, such as duplicate pin names.

Place Pin Toolbar

The buttons on the **Place Pin** toolbar found in the **Symbol Editor** are described below.

Button	Description
	Place Pin Array button. Use to place a pin array on the workspace.
	Line Pin button. Places a line pin on the workspace.
	Dot Pin button. Places a dot pin on the workspace.
	Clock Pin button. Places a clock pin on the workspace.
	Dot-Clock Pin button. Places a dot-clock pin on the workspace.
	Input Wedge Pin button. Places an input wedge pin on the workspace.
	Output Wedge Pin button. Places an output wedge pin on the workspace.
	Zero-Length Pin button. Places a zero-length pin on the workspace.

Note Refer to the Pins section for information about pin types.

Drawing Toolbar

The buttons in the **Drawing** toolbar found in the **Symbol Editor** are described below:

Button	Description
	Align Left button. Moves selected objects horizontally so that their left sides line up with the left side of the left-most object. At least two objects must be selected to enable this button. Does not work for pins.
	Align Right button. Moves selected objects horizontally so that their right sides line up with the right side of the right-most object. At least two objects must be selected to enable this button. Does not work for pins.
	Align Top button. Moves selected objects vertically so that their top sides line up with the top side of the top-most object. At least two objects must be selected to enable this button. Does not work for pins.
	Align Bottom button. Moves the selected objects vertically so that their bottom sides line up with the bottom side of the bottom-most object. At least two objects must be selected to enable this button. Does not work for pins.
	Snap To Grid button. Snaps the selected element(s) to the **Draw Grid** that is found within the symbol's boundary box. Does not work for pins.
	Distribute Horizontal button. Evenly spaces the selected objects horizontally. At least three objects must be selected to enable this button. Does not work for pins.
	Distribute Vertical button. Evenly spaces the selected objects vertically. At least three objects must be selected to enable this button. Does not work for pins.
	Bring To Front button. Brings selected element(s) to the foreground on the workspace. Other element(s) appear behind them. Does not work for pins.
	Send To Back button. Sends selected element(s) to the background on the workspace. Other elements appear in front of them. Does not work for pins.

Button	Description
	Rotate 90 Counter CW button. Rotates the selected element(s) 90 degrees counter-clockwise. Does not work for pins.
	Rotate 90 Clockwise button. Rotates the selected element(s) 90 degrees clockwise. Does not work for pins.
	Flip Horizontal button. Flips the selected element(s) horizontally on the workspace. Does not work for pins.
	Flip Vertical button. Flips the selected element(s) vertically on the workspace. Does not work for pins.
	Ungroup button. Returns an element that was made using the **Group** command back to its individual elements. Does not work for pins.
	Group button. Places selected elements in one group. Does not work for pins.
	Resize Boundary Box button. Places a cursor at the lower-right side of the boundary box. Drag it to the desired location to resize the boundary box. You cannot make the boundary box smaller than the elements that it contains or smaller than is required for the pins that are attached to it.

Editing a Component's Model

If you wish to simulate a component, it must have a model.

The following are found in the **Model** tab of the **Component Properties** dialog box:

- **Model Name** area—Displays the list of models associated with the component.
- **Model Data** area—Displays the SPICE model data of the selected component.
- **Symbol Pins** column—Found in the **Pin Mapping Table**. Displays the names of the pins associated with the symbol.

- **Model Nodes** column—Found in the **Pin Mapping Table**. Displays the order in which the **Symbol Pins** are represented in the model data.
- **Add from Comp** button—Use to select a component, whose model you wish to use, from the existing Multisim database.
- **Add/Edit** button—Use to add or edit a new or existing model in the Multisim database.
- **Delete a Model** button—Use to delete a model or number of models from the model list in the **Model Name** area.
- **Copy to** button—Use to copy model information to other sections of a multisection component.
- **Show Template** button—Use to display template of pins connected to their respective nodes in the model.

The **Model Data** and **Pin Mapping Table** areas contain the information that make up the model itself (and are thus the most important parts for simulation purposes).

Note If you modify a model, the model changes for all other components within the same database, whose models are based on that template. If you want to modify a model for a specific component, save the modified model with a name similar to that of the component. Alternatively, you can copy a model whose model template matches what you want.

Note If you select a component from the **Master Database**, you will not be able to modify its model information.

You can import or load an existing model. Refer to the *Loading an Existing Model* section for information.

Depending on your edition of Multisim, you can create your own model using the **Model Makers**. Refer to the *Creating a Component Model Using the Model Makers* section for more information.

Caution If you choose to edit a model's data or template directly, be very careful when entering information. Making a typing error or removing a character by mistake could cause the model to function improperly. Unless you are experienced at creating/editing models, it is recommended that you copy a model that has the same template information you require.

Adding a Model to a Component

Multisim allows you to create models and place them in either the **Corporate Database** or **User Database**. These models can then be used when creating components or editing a component's model.

Complete the following steps to create a model and save it to the **User Database** or **Corporate Database**:

First add a new model ID to the Database.

1. Click the **Add/Edit** button on the **Model** tab of the **Component Properties** dialog box. The **Select a Model** dialog box appears:
   - **Add** button—Adds a new model name to the selected database. Is only active for the **Corporate** or **User Database**.
   - **Delete** button—Deletes any of the models listed in the **Model ID List**. Is only active for the **Corporate** or **User Database**.
   - **Link Info** button—Displays a list of components with the selected model through the **Component List** dialog box.
   - **Copy to** button—Copies the selected model to either the **Corporate** or **User Database**.
   - **Rename** button—displays a dialog where you rename the selected model.
   - **Filter** button—Displays the **Filter** dialog box, where you select the manufacturer and model ID to display.
   - **Start Model Maker** button—Accesses Multisim's **Model Makers** to create a SPICE model for the selected model.
   - **Load Model from File** button—Use to select a model from a file in SPICE, Verilog, VHDL, or Code Model format.
2. Select the database in the **Database Name** field where you want the new model to be stored.
3. Click the **Add** button. The **Set Parameters** dialog box appears. Enter the name that you would like to call the new model and click **OK**.
4. The name of the model appears in the **Model ID List** of the database you selected.
5. Highlight the model you just created in the **Model ID List**. Notice there is no model data provided.

Next, enter information into the model; you can create a model by writing a SPICE model or using the available **Model Makers**, load a model from a file, or copy existing model data information directly into the **Model Data** field.

Component List Dialog Box

Complete the following steps to display a list of components with a selected model:

1. Select a model in the **Model ID list** in the **Select a Model** dialog box.
2. Click **Link Info** in the **Select a Model** dialog box. The **Component List** dialog box displays.
3. Click **OK** to close the dialog.

Creating a SPICE Model for a Component

Multisim is based on industry standard SPICE 3F5. It supports models created using standard SPICE syntax. You can create a model using the **Model Makers**, by assigning values to the parameters of a primitive model, or by creating a subcircuit model.

Creating a Model Using a Model Maker

Complete the following steps to use a model created by the **Model Makers**:

1. Select the **Model ID** in the **Select a Model** dialog box.
2. Click the **Start Model Maker** button. The **Select Model Maker** dialog box appears.
3. Select the **Model Maker** you wish to use to make a model.
4. Click **Accept** to continue to start the process of making a model. Click **Cancel** to return to the **Model** tab of the **Component Properties** dialog box.
5. For analog **Model Makers**, refer to the *Creating a Component Model Using the Model Makers* section for information about using specific **Model Makers**. For RF model makers, refer to the *RF Model Makers* section of Chapter 14, *RF*.
6. When you have entered all the required information in the **Model Maker** dialog box, click **OK**. The data for the model you have just created will appear in the **Model Data** field.

Creating a Primitive Model

Certain devices have primitive SPICE models. Some of these devices are listed below. A primitive model is a model that is defined by a collection of parameters. They are used as basic building blocks in circuits and in subcircuits:

- R—Semiconductor resistor model.
- C—Semiconductor capacitor model.

- SW—Voltage controlled switch.
- CSW—Current controlled switch.
- URC—Uniform distributed RC model.
- LTRA—Lossy transmission line model.
- D—Diode model.
- NPN—NPN BJT model.
- PNP—PNP BJT model.
- NJF—N-channel JFET model.
- PJF—P-channel JFET model.
- NMOS—N-channel MOSFET model.
- PMOS—P-channel MOSFET model.
- NMF—N-channel MESFET model.
- PMF—P-channel MESFET model.

Refer to Chapter 7, *Multisim SPICE Reference*, for more information.

An example of a primitive model for a 2n2222a NPN BJT Transistor follows. The first line of a primitive model begins with the statement, .MODEL, followed by the model name and then the primitive type. In the example below the model name is “2N2222A” and the primitive type is “NPN”. The rest of the lines of the model define the parameters of the NPN BJT. Note that they begin with a “+”. Details regarding the parameters are found in the *Component Reference help file*. You do not need to define all of the parameters; any that are omitted are assigned default values.

```
.MODEL 2N2222A NPN
+IS=2.04566e-13 BF=296.463 NF=1.09697 VAF=10
+IKF=0.0772534 ISE=1.45081e-13 NE=1.39296 BR=0.481975
+NR=1.16782 VAR=100 IKR=0.100004 ISC=1.00231e-13
+NC=1.98587 RB=3.99688 IRB=0.2 RBM=3.99688
+RE=0.0857267 RC=0.428633 XTB=0.1 XTI=1
+EG=1.05 CJE=1.09913e-11 VJE=0.99 MJE=0.23
+TF=2.96787e-10 XTF=9.22776 VTF=25.2257 ITF=0.0793144
+CJC=3.1941e-11 VJC=0.4 MJC=0.85 XCJC=0.901093
+FC=0.1 CJS=0 VJS=0.75 MJS=0.5
+TR=3.83883e-07 PTF=0 KF=0 AF=1
```

For further information regarding primitive models, please consult the component *Component Reference help file* or the SPICE 3F5 user manual (`http://bwrc.eecs.berkeley.edu/Classes/IcBook/SPICE`).

Creating a SPICE Subcircuit

Many electronics devices are not represented by primitives, but are still well suited as SPICE models. SPICE subcircuit are used to capture the characteristics of these models. A SPICE subcircuit is built from a collection of devices that contain primitive models, voltage and/or current sources, and/or other SPICE subcircuits.

You can either create a SPICE subcircuit from scratch by typing it into the model data window or you can first draw the circuit in Multisim and export the SPICE netlist and then modify it for use as a SPICE subcircuit.

All SPICE subcircuits must begin with a line that begins with .SUBCKT followed by the SPICE subcircuit name and the external nodes of the SPICE subcircuit that will be connected to other components. The SPICE subcircuit must end with the statement .ENDS.

```
.SUBCKT <SubcircuitName> <N1> <N2> <N3> <N4>
…
.ENDS SubcircuitName
```

A SPICE subcircuit is defined by naming and connecting the internal devices that make up the subcircuit. For example, to define that a 100kΩ resistor with the reference designator R1 is connected between nodes 4 and 5 you would write:

```
R1 4 5 100k
```

An example of a SPICE subcircuit follows:

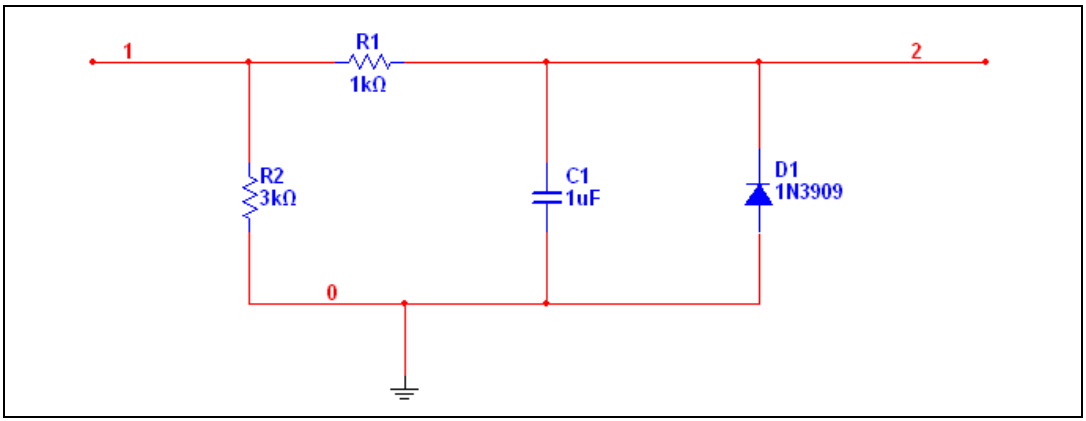
```
.SUBCKT SampleSubcircuit 4 2

R1 1 2 1.000e+003
C1 2 0 1.0E-6
R2 1 0 3.0k
D1 0 2 D1N3909


.MODEL D1N3909 D (
+    IS =  6.233e-10
+    RS =  0.003866
+    CJO =  1.132e-10
+    VJ =  0.75
+    TT =  3.699e-07
+    M =  0.2711
+    BV =  100
+    N =  1.505
+    EG =  1.11
```

```
+    XTI =  3
+    KF =   0
+    AF =   1
+    FC =   0.5
+    IBV =  0.0001
+    TNOM =  27
+ )
.ENDS
```

This is the subcircuit for the following circuit drawn in Multisim:



Note that nodes 1 and 2 are the two nodes that will be connected outside of the subcircuit model. They are defined as external nodes by including them in the first line of the subcircuit model.

The resistors, capacitor, and diode are all primitives. The only parameters defined for the resistors and capacitor are their values, but the diode has additional parameters defined using the `.model` statement.

Refer to Chapter 7, *Multisim SPICE Reference*, for more information.

Loading an Existing Model

Complete the following steps to load or import an existing VHDL, Verilog, Ccode model, or SPICE model for your component:

1. Click **Load Model From File** in the **Select a Model** dialog box. A standard file browser appears, allowing you to choose the file type, name and location of the files to be loaded. Before loading a netlist, make sure you know what folder it is in. Most Bspice, Xspice and Cadence® PSpice® netlists end in extensions `.cir` or `.net`.

2. Select the file to load and click **OK**. The model data appears in the **Model** tab fields of the **Component Properties** dialog box.

Note Cadence® PSpice® is not an industry standard, but is proprietary to the Orcad SPICE simulation tool. Since some component vendors make models for their components available in Cadence® PSpice® format, Multisim has been designed to support Cadence® PSpice® models as extensively as possible. However, you will not be able to share models or circuits with other SPICE users or tools. Refer to the *Compatibility Modes* section of Chapter 7, *Multisim SPICE Reference*, for more information.

Modify a Model's Data

Models can only be modified if they are in the **Corporate** or **User** database. Multisim does not allow information stored in the **Master Database** to be edited.

Complete the following steps to modify a model in the **Corporate** or **User** database:

1. Select the model you wish to edit by clicking on it in the **Model ID List** in the **Select a Model** dialog box.
2. The model data can be modified by making changes directly in the **Model Data** area of the **Select a Model** dialog box.
3. Select the **Save** button to save any changes that were made to the model.

Note If the model that has been modified is being used by a component in the database where it is located (**Corporate** or **User**) a message window will appear indicating that the model has been linked to a component in the component database. Multisim will not allow you to save the modifications to this named model. You will have to change the name of the model.

Copying the Model of One Component to Another

Complete the following steps to copy the model information from an existing component:

1. Click **Add From Comp** in the **Model** tab of the **Component Properties** dialog box. The **Select Model Data** dialog box appears.
2. Select the database you want to choose the model from in the **Database** drop-down list.
3. Select the **Group** and **Family** of the component you want to choose the model from.

4. In the **Component** list, choose the component whose model template most closely matches what you want. The model's manufacturer and footprint type appear in the **Model manuf./ID** and **Footprint manuf./Type** fields.
5. To view the model of the component you selected, click the **Model** button.
6. Click **OK** to return to the **Component Properties** dialog box.
7. The model name is added to the list of models associated with the component.

Analog Behavioral Modeling and Controlled Source Syntax

Analog Behavioral Modeling (ABM) is an extremely powerful feature which provides an efficient way to macro model signal processes through non-linear mathematical and conditional expressions. For instance, if you require a comparator for a PWM based circuit, instead of modeling the inner working of the comparator, the simple `"if(Vin1 > Vin2, 10, 0)"` expression can be used. ABM-based voltage and current sources are based on mathematical expressions of circuit voltages and currents. The supported functions, operators, constants, and syntax are described in this section.

Accessing net voltages and branch currents in ABM expressions

Expressions may contain net voltages relative to ground using the syntax `V(<net name>)`, where `<net name>` is the name of a net. Alternatively, expressions may contain differential net voltages using the syntax `V(<net_name1>, <net_name2>)`, where the expression evaluates to the difference between nets `<net name1>` and `<net name2>`.

For example:

```
B1 net1 net2 v={V(3)+V(7,2)}
```

Expressions may contain branch currents through voltage source elements using the syntax `I(<refdes>)`, where `<refdes>` is the reference designator of the voltage source. If a branch current is required that does not contain an appropriate voltage source, simply break the branch and insert a 0-volt voltage source. Positive current polarity is taken from the positive net to the negative net of the voltage source.

Multisim supports referencing current through five types of SPICE primitive voltage sources: Independent source (“V” source), Voltage Controlled Voltage Source (“E” source), Current Controlled Voltage Source (“H” source), the ABM source (“B” source), and Inductors.

For example:

```
B1 net1 0  v={ 5+ I(V2)+3*I(E5)-2.2*I(B2)}
```

Editing a Component Pin Model

The **Pin Parameters** tab of the **Component Properties** dialog box is where you assign the pin type for each of the symbol pins in a component.

Note **Symbol Pin** is the name of the pin. For example, most digital parts have pins named “Vcc” and “GND”.

Complete the following steps to enter pin parameters:

1. Select the desired **Symbol Pin**.
2. Select the type of component in the **Component Type** drop-down list.
3. Select the technology in the **Component Technology** drop-down list.
4. For each symbol pin, click in the corresponding field in the **Type** column and select the pin type from the drop-down list that appears.

Tip You can assign the same pin type to multiple symbol pins at once. Use the <Ctrl> and <Shift> keys to select the desired pins and then select the desired type from one of the *selected* **Type** fields. The type for all of the selected symbol pins changes to reflect your selection.

5. Select whether or not you wish to include each symbol pin in the **Electrical Rules Check** by clicking in the corresponding field in the **ERC Status** column and selecting either **Include** or **Exclude**. (The default is **Include.**)

Note The **Section** column displays the section of a multi-section component. This column is read-only.

Editing a Component's Footprint

The **Footprint** tab of the **Component Properties** dialog box allows you to specify/modify package information for a component and map footprint and symbol pins of a component:

- **Footprint Manufacturer\Type** area—Shows the manufacturer and footprint name associated with the component.
- **Symbol Pin to Footprint Pin Mapping Table**—Shows the symbol pin names of the component and the footprint pins they are associated with in the package.
- **Add From Database** button—Use to add a footprint package to the component from a list of footprints in the selected database.
- **Delete** button—Use to delete any of the footprint packages assigned to the component in the **Footprint Manufacturer\Type** area.
- **Change** button—Displays the **Change Footprint** dialog box. Refer to the *Edit Footprint Dialog Box* section of Chapter 3, *Schematic Capture - Advanced Functions*, for information.
- **Map Pins** button—Displays the **Advanced Pin Mapping** dialog box.

Note A footprint in the **Master Database** cannot be modified. To modify a footprint a copy must be added to the **Corporate Database** or **User Database**.

Select a Footprint dialog box

Complete the following steps to modify or enter package information:

1. Click the **Add From Database** button in the **Footprint** tab of the **Component Properties** dialog box. The **Select a Footprint** dialog box appears:
   - **Database Name** list—The database the footprints are in.
   - **Data List**—Shows the footprints in the selected database.

Note When you create or edit a part with a custom land pattern (footprint) in Ultiboard, the change is automatically synchronized in Multisim. This appears in the preview field and the **Footprint** column (if the footprint name is changed) of the **Select a Footprint** dialog box. If the database containing the part is already open in the **Select a Footprint** dialog box in Multisim when you create or edit the part in Ultiboard, you must select a different database and then reselect the original database to view the change.

- **Filter** button—Use to filter the contents of the **Data List**. Refer to the *Filter dialog box* section for more information.
- **Add** button—Use to add a footprint to the **User** or **Corporate Database**. Refer to the *Add a Footprint dialog box* section for more information.
- **Delete** button—Use to delete a footprint from the **Corporate** or **User Database**.
- **Copy to** button—Use to copy a footprint from one database to another.
- **Link Info** button—Displays a list of components with the selected footprint through the **Component List** dialog box.

2. Select the **Database Name** where the footprint is that you want to edit.
3. If the footprint is in the **Master Database**, select the footprint in the **Data List** and click the **Copy To** button. The **Set Parameters** dialog box appears prompting you to place it in the **Corporate Database** or **User Database**.
4. Click **OK**. The footprint is copied to the database you indicated and you are returned to the **Select a Footprint** dialog box.
5. Click the field of the footprint you want to modify.
6. Enter the appropriate information.

Note Once a footprint is modified, you will be prompted to save the footprint with the modifications. If you choose not to save, the modifications will be lost.

7. Click the **Save** button. The modifications are saved.

Complete the following steps to map symbol and footprint pins:

1. Navigate to the **Footprint** tab of the **Component Properties** dialog box.
2. Select the footprint you wish to make modifications to in the **Footprint Manufacturer\Type** box.
3. In the **Symbol Pin to Footprint Pin Mapping Table**, for each **Symbol Pin**:
   - Select its corresponding footprint pin from the drop-down list in the **Footprint Pins** column, using information from the component's datasheet.

Note Refer to the *Pins* section for information about the differences between symbol pins and footprint pins.

- Select its swap group from the drop-down list in the **Swap Group** column. Pins in the same swap group can be interchanged during the PCB layout process.

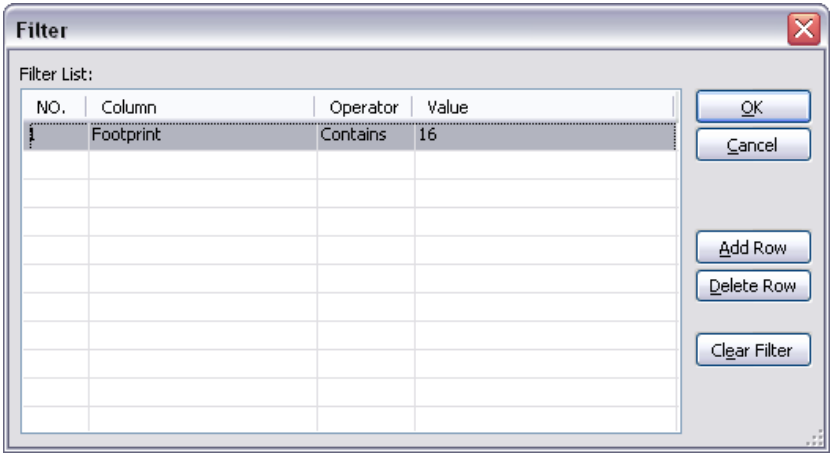
Filter dialog box

You can filter what is displayed in the **Data List** of the **Select a Footprint** dialog box by using the **Filter** dialog box. For example, you may wish to view a specific footprint from a specific device manufacturer only.

Complete the following steps to filter contents of the **Data List**:

1. Click **Filter** in the **Select a Footprint** dialog box to display the **Filter** dialog box.
2. Click **Add Row**.
3. Enter desired parameters as shown below:
   - **Column**—Corresponds to the columns in the **Select a Footprint** dialog box. Click in the field and select the desired value from the drop-down list that appears (“Footprint”, for example).
   - **Operator**—Select one of: **=**, **NOT =**, **Contains**, **Starts with**.
   - **Value**—A value you would expect to see in the selected **Column** (“16”, for example).

   The dialog box will resemble the following:



4. Click **OK**. The **Filter** dialog box closes and the **Select a Footprint** dialog box displays with the **Data List** populated according to the parameters you entered in the **Filter** dialog box.

Note **Filter On** remains visible until **Clear Filter** is clicked in the **Filter** dialog box.

5. Optionally, click **Filter** to display the **Filter** dialog box and enter more parameters.
6. Click **Add Row** and enter parameters as shown above.
7. Click **OK** to return to the **Select a Footprint** dialog box. The contents of the dialog now reflect the two filters that you entered in the **Filter List**.

Tip Add as many filters as necessary to reduce the items in the **Data List** to an easily scrollable quantity.

To display all available footprints in the **Select a Footprint** dialog box, click **Clear Filter** and **OK** from the **Filter** dialog box.

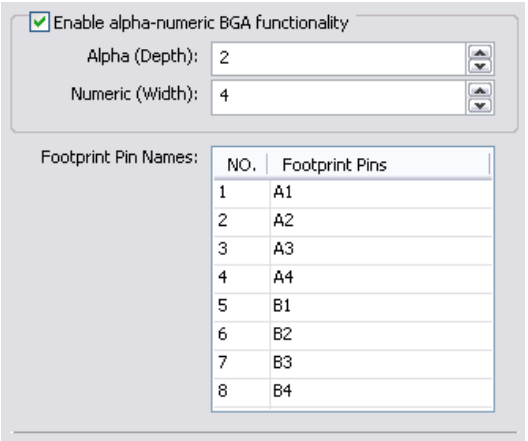
To clear an individual filter from the **Filter** dialog box, select the filter and click **Delete Row**.

Add a Footprint dialog box

Complete the following steps to add a footprint to the **User Database** or **Corporate Database**:

1. Select either **Corporate Database** or **User Database** in the **Database Name** field of the **Select a Footprint** dialog box. Refer to the *Select a Footprint dialog box* section for more information.
2. Click on **Add** in the **Select a Footprint** dialog box. The **Add a Footprint** dialog box displays.
3. Enter the required information in the **Database Name**, **Manufacturer**, **Footprint**, **Ultiboard Footprint**, and **Number of Pins** fields.
4. Select either **SMT** (surface mount) or **TH** (through-hole) in the **SMT/TH** field.
5. For BGAs (Ball Grid Arrays):
   - Enable the **Enable alpha-numeric BGA functionality** checkbox.
   - **Alpha (Depth)**—Enter the number of rows of footprint pins in the BGA.
   - **Numeric (Width)**—Enter the number of columns of footprint pins in the BGA.

As you change the values in the above two fields, the **Footprint Pins** values are updated as in the example shown below.



Note BGA pins in Ultiboard (Circuit Design Suite's PCB layout software), are named in a matrix array starting with A1, A2…Ax; the next row would be B1, B2...Bx, etc. When creating this symbol in Multisim, the footprint pins are initially named 1, 2, 3 etc… . The functionality described in this step allows the footprint pins to be renamed to match the alpha-numeric pin naming method used in Ultiboard. This feature is especially useful for BGAs with large numbers of pins that would otherwise have to be manually renamed.

6. Click **OK**.

Advanced Pin Mapping Dialog

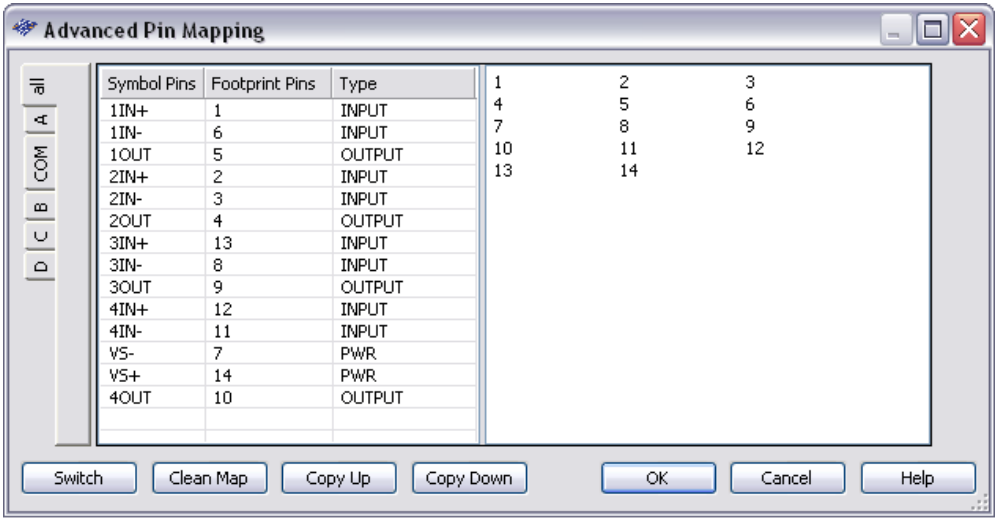
The **Advanced Pin Mapping** dialog box is used to map symbol pins to footprint pins and is especially useful for more complex components.

Complete the following steps to display the **Advanced Pin Mapping** dialog box:

1. Click the **Map Pins** button in either the **Footprint** tab of the **Component Properties** dialog box, step 5 of the **Component Wizard**, or the **Edit Footprint** dialog box.

   This example shows the pin mapping for an LM3301N, Quad High Gain Norton Op-amp.

   The all tab shows all pins in the component. Tabs A, B, C & D show pins from individual sections. The COM tab shows pins common to all sections. The footprint pins display in the preview pane on the right side of the dialog box.



To reverse the display so that the symbol pins appear in the right pane, click **Switch**.

To remove all pin mapping, click **Clean Map**.

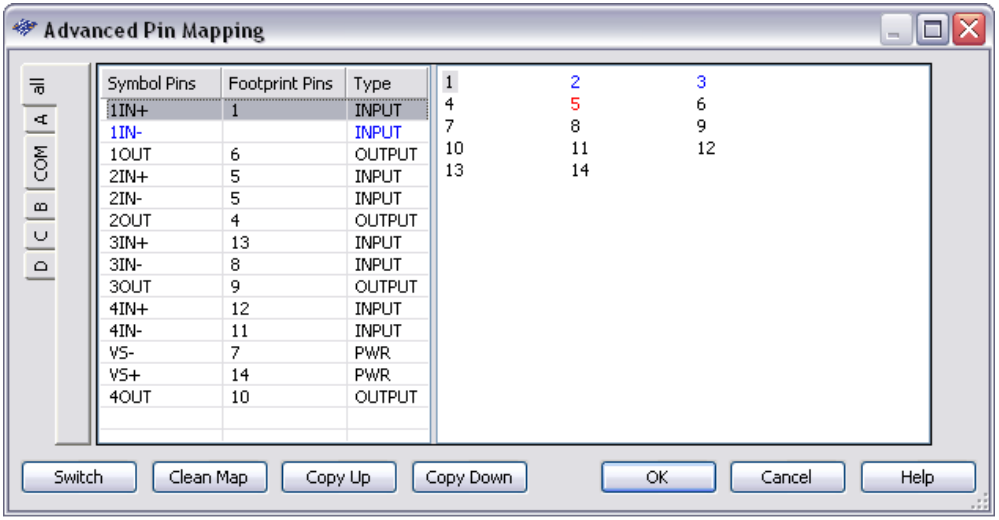
Mapping Pins

The following example uses the **Advanced Pin Mapping** dialog box with the footprint pins displayed in the right pane. If you wish to display the symbol pins in the right pane, click **Switch**. The same mapping techniques will apply.

Complete the following steps to map footprint to symbol pins:

1. Display the **Advanced Pin Mapping** dialog as described earlier in this section.

   In the preview pane, a black footprint pin number indicates that the footprint pin has been mapped to a symbol pin; a blue footprint pin number indicates an unmapped footprint pin; a red footprint pin number indictes a mapping duplication; a grey footprint pin number indicates that that pin has been selected in the table on the left, as with footprint pin 1 in the example below.



Tip Hover the cursor over a pin in the right pane to display a tooltip that shows the pin's mapping.

2. Select a symbol pin in the left column and click on a blue (unmapped) footprint pin in the right pane. The footprint pin's color turns black to indicate that it is mapped.
3. You can also click-and-drag a symbol pin to footprint pin in the right pane to create a mapping (or vice versa).

Complete the following steps to use the **Copy Up** and **Copy Down** buttons:

1. Use the <Shift> and/or <Ctrl> keys to select a number of pins symbol pins in the left column.
2. Click **Copy Up** to copy the selected pin assignments upwards.

   Or

   Click **Copy Down** to copy the selected pin assignments downward.

Editing a Component's Electronic Parameters

The **Electronic Param** tab consists of two sets of fields. The **Device Specific Parameters** vary depending on the type of component.

The **Common Parameters** are common to all components. These are:

- **Thermal Resistance Junction**—Enter or modify the thermal characteristics within the component (from the junction to the case), in watts or degrees centigrade.
- **Thermal Resistance Case**—Enter or modify the thermal characteristics of the whole package (component) in watts or degrees centigrade.
- **Power Dissipation**—Enter or modify the power dissipation of the component, in watts.
- **Derating Knee Point**—Enter or modify the temperature (degrees centigrade) at which the power of the component/package begins to be de-rated, in order to operate the device in its safe operating range.
- **Min. Operating Temperature**—Enter or modify the lowest ambient temperature at which the component can operate reliably in degrees centigrade.
- **Max. Operating Temperature**—Enter or modify the highest ambient temperature at which the component can operate reliably in degrees centigrade.
- **ESD Rating**—Enter or modify the electro-static discharge rating for the component.

Editing User Fields

User fields can be used for any purpose you wish. For example, you might use these fields to record the cost of a component (the price you pay to the supplier or vendor), lead time for ordering, preferred supplier, stock number, and so on. The information is particularly useful in reports and in searching the database for the most appropriate component.

User fields consist of two elements: the field title and the field value. Field titles are shared across databases (that is, all components in all the databases have the same user field titles), and field values are unique to an individual component. Typically, a manager or project leader would populate this level of the database.

Note User fields must be given titles before they can be used to record values.

Complete the following steps to set up or modify user field titles:

1. Click the **Database Manager** button from the **Standard** toolbar.

   Or

   Choose **Tools»Database»Database Manager**.
2. Click the **User Field Titles** tab.
3. Click the appropriate **Title** until a frame appears around it.
4. Enter or change the name for the **Title**. This name will appear in the **Component Properties** dialog box for all components in all databases.

Creating a Component Model Using the Model Makers

Multisim offers advanced **Model Makers** which automatically generate simulation models for you when you give them databook values as input. **Model Makers** will save you time and effort but do require experience for you to become proficient with them.

For each **Model Maker**, preset values are provided for a specific model. However, these are not default values, and you can select numerical values based on the component you are interested in, using data from a databook.

When working with databooks, note that different databooks provide parameters for a component model in different styles. While some information is given numerically in tables or lists for a specific operating point, other information is given in the form of a chart or graph. Both types of information are required by Multisim's **Model Makers**. In the case of tables or lists, you will need to enter the operating point and the value that you want. In the case of charts or graphs, the way you select the points from the appropriate curves will have an impact on the accuracy of the parameters of the final model. This guide gives suggestions on methods for selecting points in the procedures for the **Model Makers**. Also, note that the information provided by databooks is usually the same from one manufacturer to another, even though the names or labels and descriptions of parameters are different.

AC Motor

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.

3. From the **Model Maker List**, select **AC Motor** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **3 Phase AC Motor** dialog box appears.
4. Enter desired values in the **3 Phase AC Motor** dialog box.
5. When all values are entered, click **OK** to complete the model.

BJT Model Maker

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **BJT** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **BJT Model** dialog box appears.
4. Enter values in the **BJT Model** dialog box as described in the sections below.
5. When all values are entered, click **OK** to complete the model, or click **Cancel** to cancel.

Note The **BJT Model** dialog box shows preset values for the MPS2222 model.

Entering General and Table Data

1. In the **BJT Model** dialog box, click the **General** tab.
2. Locate data information for the BJT model from a databook.

Complete the following steps to enter **General** data:

1. Enter **NPN** or **PNP** in the **Type of BJT** field. This is usually found on the first page of the data book.
2. In the **Type of Semiconductor** field, enter the type of semiconductor. This is usually found written next to the component type.
3. If desired, change the default value set by Multisim for **Nominal Temperature**.
4. If desired, change the default value for **Base Temperature for Input**. This is typically found in the top left corner of the “Electrical Characteristics” table in the databook.

Complete the following steps to enter **Maximum Ratings** data:

1. In the databook for the BJT, locate the "Maximum Ratings" table—for example:

Enter this information in the **Emitter-Base Maximum Voltage** field.

MAXIMUM RATINGS

Rating	Symbol	MPS2222	MPS2222A	Unit
Collector–Emitter Voltage	V_{CEO}	30	40	Vdc
Collector–Base Voltage	V_{CBO}	60	75	Vdc
Emitter–Base Voltage	V_{EBO}	5.0	6.0	Vdc
Collector Current — Continuous	I_C	600		mAdc
Total Device Dissipation @ T_A = 25°C Derate above 25°C	P_D	625 5.0		mW mW/°C
Total Device Dissipation @ T_C = 25°C Derate above 25°C	P_D	1.5 12		Watts mW/°C
Operating and Storage Junction Temperature Range	T_J, T_{stg}	−55 to +150		°C

2. Find the value for Emitter-Base Voltage and enter the value in the **Emitter-Base Maximum Voltage (VEBO)** field.

Complete the following steps to enter **Output Admittance** data:

1. In the databook, locate the "Small Signal Characteristics" table, and find the values for Output Admittance—for example:

Use this information to enter data in the **Output Admittance** fields.

SMALL–SIGNAL CHARACTERISTICS

Current–Gain — Bandwidth Product(2) (I_C = 20 mAdc, V_{CE} = 20 Vdc, f = 100 MHz)	MPS2222 MPS2222A	f_T
Output Capacitance (V_{CB} = 10 Vdc, I_E = 0, f = 1.0 MHz)		C_{obo}
Input Capacitance (V_{EB} = 0.5 Vdc, I_C = 0, f = 1.0 MHz)	MPS2222 MPS2222A	C_{ibo}
Input Impedance (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) (I_C = 10 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	MPS2222A MPS2222A	h_{ie}
Voltage Feedback Ratio (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) (I_C = 10 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	MPS2222A MPS2222A	h_{re}
Small–Signal Current Gain (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) (I_C = 10 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	MPS2222A MPS2222A	h_{fe}
Output Admittance (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) (I_C = 10 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	MPS2222A MPS2222A	h_{oe}

If data are not available, enable **Check if data not available**.

2. Based on the table data, enter:
   - **Output Admittance (hoe)**
   - **Collector Current (Ic1)**
   - **Collector-Emitter Voltage (Vce)**

Note Databooks provide maximum and minimum values for the **Output Admittance** parameter. Select a typical value of output admittance.

Complete the following steps to enter **Switching Characteristics** data:

1. In the databook, find the "Switching Characteristics" table—for example:

Use this information to enter data in the **Switching Characteristics** fields.

SWITCHING CHARACTERISTICS MPS2222A only

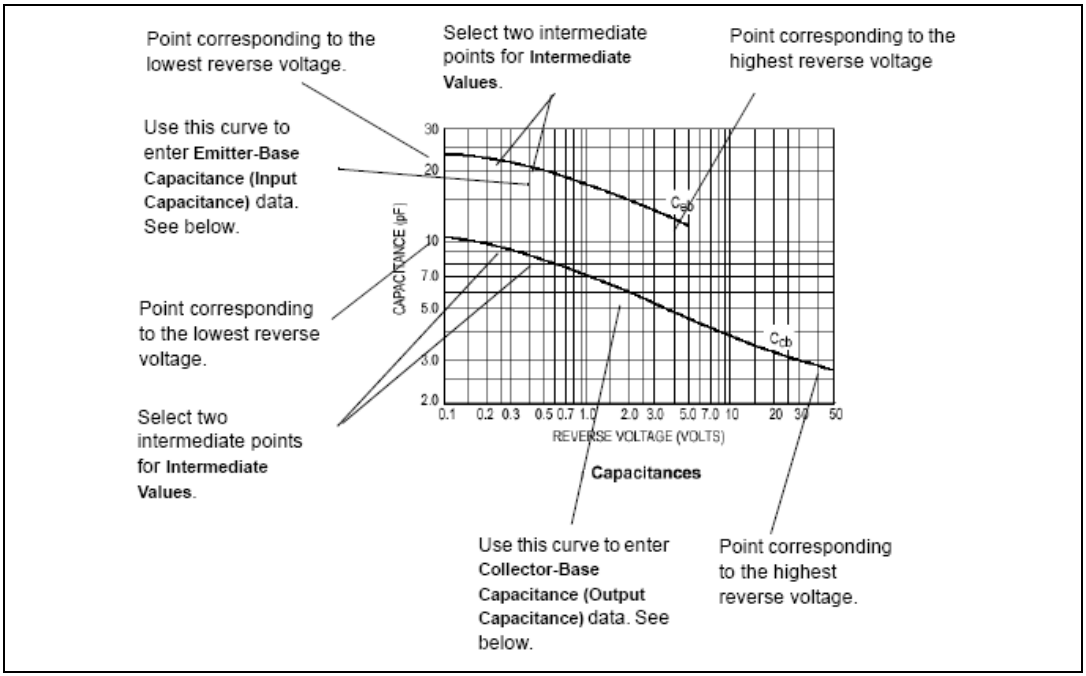
Delay Time	(V_{CC} = 30 Vdc, $V_{BE(off)}$ = −0.5 Vdc, I_C = 150 mAdc, I_{B1} = 15 mAdc) (Figure 1)	t_d
Rise Time		t_r
Storage Time	(V_{CC} = 30 Vdc, I_C = 150 mAdc, I_{B1} = I_{B2} = 15 mAdc) (Figure 2)	t_s
Fall Time		t_f

1. Pulse Test: Pulse Width ≤ 300 μs, Duty Cycle ≤ 2.0%.
2. f_T is defined as the frequency at which $|h_{fe}|$ extrapolates to unity.

2. Based on the table data, enter:
   - **Storage Time (ts)**
   - **Collector Current (Ic2)**
   - **Base Current (Ib1)**
   - **Base Current (Ib2)**

Entering Capacitances Data

1. Click the **Capacitances** tab.
2. In the databook, locate the "Ceb and Ccb vs. Reverse Voltages (RV)" graph—for example:


Point corresponding to the lowest reverse voltage.
Select two intermediate points for Intermediate Values.
Point corresponding to the highest reverse voltage
Use this curve to enter Emitter-Base Capacitance (Input Capacitance) data. See below.
Point corresponding to the lowest reverse voltage.
Select two intermediate points for Intermediate Values.
CAPACITANCE (pF)
30
20
10
7.0
5.0
3.0
2.0
0.1 0.2 0.3 0.5 0.7 1.0 2.0 3.0 5.0 7.0 10 20 30 50
REVERSE VOLTAGE (VOLTS)
Ceb
Ccb
Capacitances
Use this curve to enter Collector-Base Capacitance (Output Capacitance) data. See below.
Point corresponding to the highest reverse voltage.

Complete the following steps to enter **Emitter-Base Capacitance (Input Capacitance)** data:

1. On the Ceb curve, locate the point corresponding to the lowest voltage, or the beginning point, of the Ceb curve. Use the coordinates of this point to enter values for:
   - **Capacitance (Ceb1)**
   - **Low-Value of Reverse Voltage**
2. On the same curve, locate the point corresponding to the maximum voltage, or the end point. Use the coordinates of this point to enter values for:
   - **Capacitance (Ceb4)**
   - **Hi-Value of Reverse Voltage**

3. To enter **Intermediate Values**, select two intermediate points close to the left side in the low voltage region. Ensure that they are not too close, to avoid excessive error in the model. Use the coordinates of the first and second points to enter values for:
   - **Capacitance (Ceb2) at Reverse Voltage**
   - **Capacitance (Ceb3) at Reverse Voltage**

Complete the following steps to enter **Collector-Base Capacitance Chart (Output Capacitance)** data:

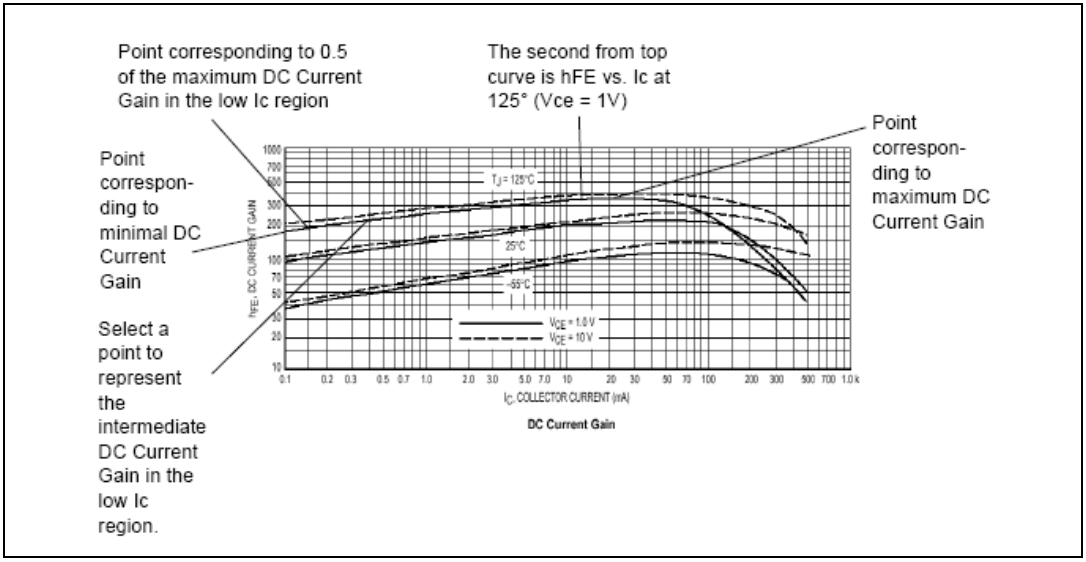
1. Using the Ccb curve from the same "Ceb and Ccb vs. Reverse Voltages (RV)" graph, repeat steps 1 through 3 above to enter values for:
   - **Capacitance (Ccb1)**
   - **Low-Value of Reverse Voltage**
   - **Capacitance (Ccb2) at Reverse Voltage**
   - **Capacitance (Ccb3) at Reverse Voltage**
   - **Capacitance (Ccb4)**
   - **Hi-Value of Reverse Voltage**

Entering DC Current Gain Chart data

1. Click the **DC Current Gain Chart (hFE vs. Ic)** tab.
2. In the databook for the BJT, locate the h_{FE} vs. I_c graph.

Complete the following steps to enter **DC Current Gain (hFE) at base Temperature** data:

1. Among the hFE vs. Ic curves at the base temperature for the BJT, select the one whose Vce is most likely the operating point for the transistor. For example:


Point corresponding to 0.5 of the maximum DC Current Gain in the low Ic region
The second from top curve is hFE vs. Ic at 125° (Vce = 1V)
Point corresponding to minimal DC Current Gain
Point corresponding to maximum DC Current Gain
Select a point to represent the intermediate DC Current Gain in the low Ic region.
DC Current Gain

Note You must select a curve with the same voltage as the Ic-Vbe curve you will use to enter data on the last tab of this dialog box. Refer to the Entering "On" Voltages and Current-Gain Bandwidth Data section for more information.

2. Find the point on the curve corresponding to the minimal collector current, or the beginning point of the curve. Use the coordinates of this value to enter:
   - **DC Current Gain (hFE1)**
   - **Minimal Collector Current**
3. Select a point from the low Ic region of the same curve. Use the coordinates of this point to enter:
   - **DC Current Gain (hFE2)**
   - **Intermediate Collector Current (low values range)**
4. Find the highest point on the curve, and enter its DC Current Gain value in the **Max Value of DC Current Gain (hFE_Max)** field.

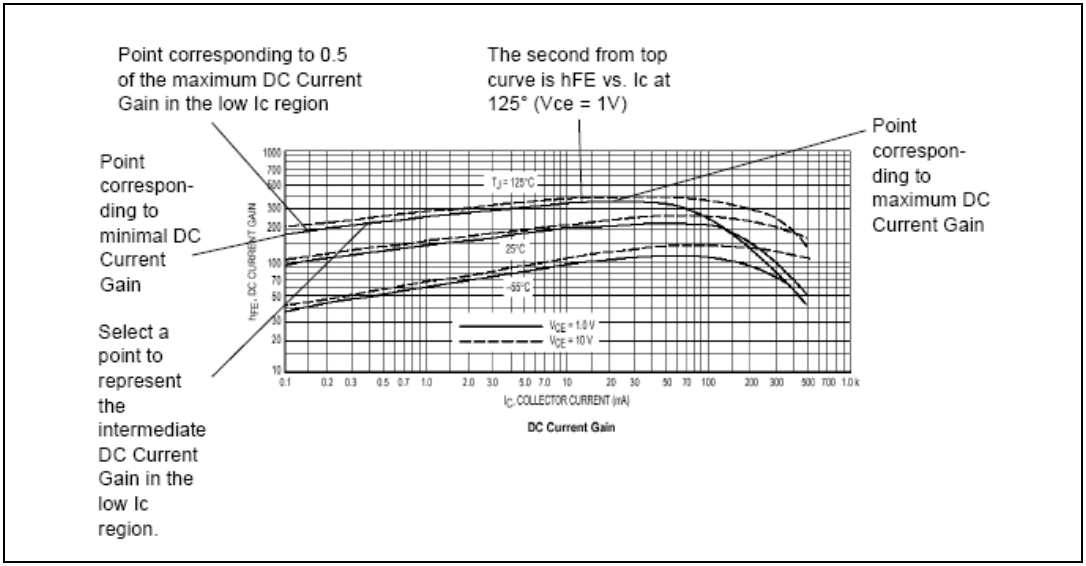
Note You must note the Ic value of this point to plot points on the Ic-Vbe curve you will use to enter data on the last tab of this dialog box. Refer to the Entering "On" Voltages and Current-Gain Bandwidth Data section for more information.

5. Find the two points corresponding to 0.5 of the maximum DC current gain value, one in the low Ic region and one in the high Ic region. Use these points to enter:
   - **Collector Current (IL) at 0.5 Max DC Current Gain (low values range)**
   - **Collector Current (Ikf) at 0.5 Max DC Current Gain (high values range)**

Note You must note the Ic value of the point in the low I_c region to plot points on the Ic-Vbe curve you will use to enter data on the last tab of this dialog box. Refer to the *Entering "On" Voltages and Current-Gain Bandwidth Data* section for more information.

Complete the following steps to enter **DC Current Gain (hFE) at another Temperature** data:

1. Using the hFE vs. Ic graph, find a curve at a different temperature from the base temperature. (This can be any other temperature.) For example:


Point corresponding to 0.5 of the maximum DC Current Gain in the low Ic region
The second from top curve is hFE vs. Ic at 125° (Vce = 1V)
Point corresponding to maximum DC Current Gain
Point corresponding to minimal DC Current Gain
Select a point to represent the intermediate DC Current Gain in the low Ic region.
hFE, DC CURRENT GAIN
IC, COLLECTOR CURRENT (mA)
TJ = 125°C
25°C
-55°C
VCE = 1.0 V
VCE = 10 V
DC Current Gain

2. Enter the temperature of the selected curve in the **Another temperature on the Chart (t2)** field.

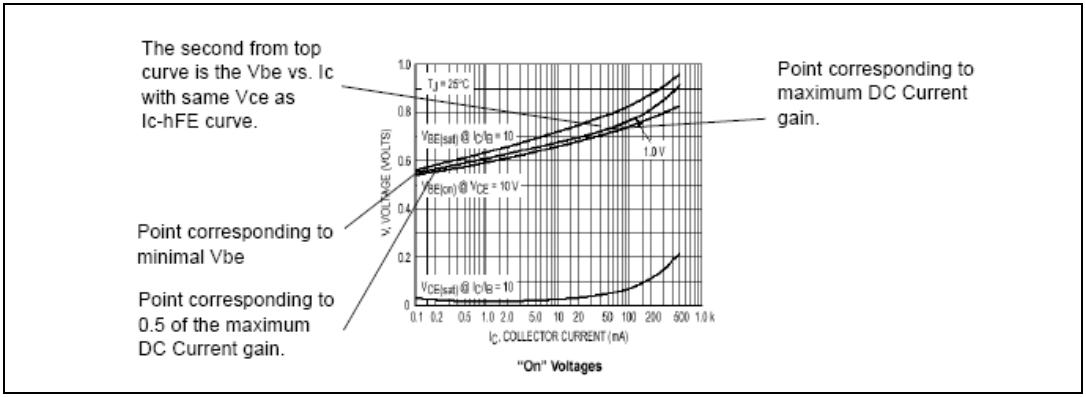
3. Find the point on the curve corresponding to the minimal collector current, or the beginning point of the curve. Use the coordinates of this value to enter:
   - **DC Current Gain (hFE1_t2)**
   - **Minimal Collector Current**
4. Select a point from the low Ic region of the same curve. Use the coordinates of this point to enter:
   - **DC Current Gain (hFE2_t2)**
   - **Intermediate Collector Current (low values range)**
5. Find the highest point on the curve, and enter its DC Current Gain value in the **Max Value of DC Current Gain (hFE_Maxt2)** field.
6. Find a point corresponding to 0.5 of the maximum DC current gain value in the low Ic region and enter its value in the **Collector Current (IL_t2) at 0.5 Max DC Current Gain (low values range)** field.

Entering "On" Voltages and Current-Gain Bandwidth Data

1. Click the **"On" Voltages, Current-Gain Bandw** tab.
2. In the databook, locate the Ic vs. Vbe graph.

Complete the following steps to enter **DC Current Gain (hFE) at Base Temperature** data:

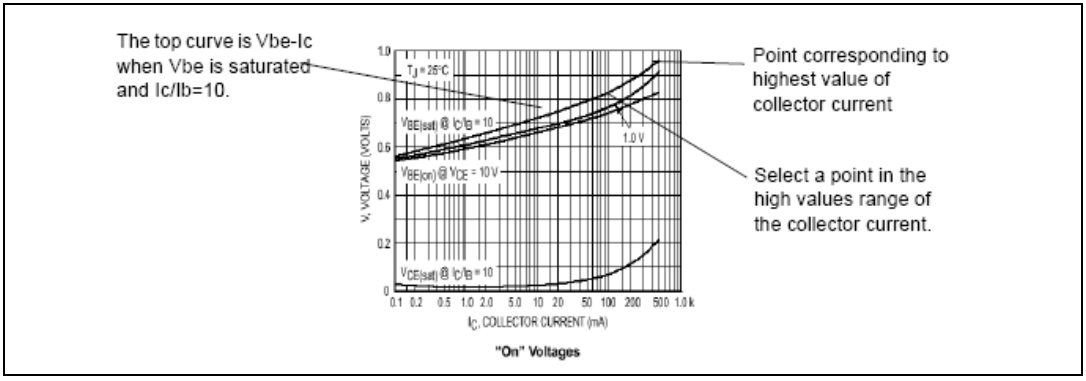
1. On the graph, locate the curve with the same Vce as the one used in the hFE data. Enter its Vce value in the **Collector-Emitter Voltage for Vbe vs. Ic (same as hFE curve)** field. For example:


The second from top curve is the Vbe vs. Ic with same Vce as Ic-hFE curve.
Point corresponding to minimal Vbe
Point corresponding to 0.5 of the maximum DC Current gain.
Point corresponding to maximum DC Current gain.
1.0
0.8
0.6
0.4
0.2
0
V, VOLTAGE (VOLTS)
TJ = 25°C
VBE(sat) @ IC/IB = 10
1.0 V
VBE(on) @ VCE = 10 V
VCE(sat) @ IC/IB = 10
0.1 0.2 0.5 1.0 2.0 5.0 10 20 50 100 200 500 1.0 k
IC, COLLECTOR CURRENT (mA)
"On" Voltages

2. Find the point on this curve corresponding to the minimal Ic value, or the beginning point of the curve. Use the coordinates of this point to enter:
   - **"On" Base-Emitter Voltage (Vbe1)**
   - **Low-Value of Collector Current**
3. Using the Ic vs. hFE graph from the previous section, locate the Ic-hFE curve at the base temperature that was used to enter data on the third tab of this dialog box. At the point of the maximum DC Current Gain (hFE), note the coordinate for the collector current (Ic).
4. On the Ic-Vbe graph, find the point corresponding to this coordinate for Ic on the curve used in steps 1 to 3. Enter the voltage for this point in the **"On" Base-Emitter Voltage (Vbe_hFEMax) at Max Gain** field.
5. Using the Ic vs. hFE graph from the previous section, locate the Ic-hFE curve at the base temperature that was used to enter data on the third tab of this dialog box. At the point corresponding to 0.5 of the maximum DC Current Gain (hFE), note the coordinate for the collector current (Ic).
6. On the Ic-Vbe graph, find the point corresponding to this coordinate for Ic on the curve used in steps 1 to 4. Enter the voltage for this point in the **"On" Base-Emitter Voltage (Vbe_iL) at 0.5 Max Gain Collector Current (low values range)** field.

Complete the following steps to enter **Vbe(sat)-Ic** data:

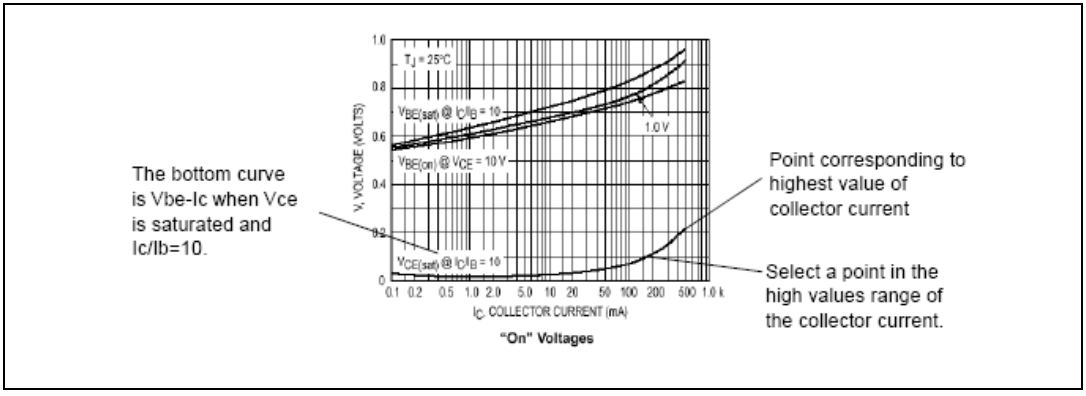
1. Using the Ic vs. Vbe graph, locate the curve whose Vbe(Sat)@Ic/Ib=10. For example:


The top curve is Vbe-Ic when Vbe is saturated and Ic/Ib=10.
Point corresponding to highest value of collector current
Select a point in the high values range of the collector current.
T_J = 25°C
$V_{BE(sat)}$ @ Ic/Ib = 10
$V_{BE(on)}$ @ V_{CE} = 10 V
$V_{CE(sat)}$ @ Ic/Ib = 10
1.0 V
V, VOLTAGE (VOLTS)
1.0
0.8
0.6
0.4
0.2
0
0.1 0.2 0.5 1.0 2.0 5.0 10 20 50 100 200 500 1.0 k
I_C, COLLECTOR CURRENT (mA)
"On" Voltages

2. Find the highest point on the curve. Use the coordinates of this point to enter:
   - **Saturation Base-Emitter Voltage (Vbe2_sat)**
   - **Hi-Value of Collector Current**
3. Select a point on the curve in the high values range of the collector current. Use the coordinates of this point to enter:
   - **Saturation Base-Emitter Voltage (Vbe1_sat)**
   - **Collector Current in the high values range**

Complete the following steps to enter **Vce(sat)-Ic** data:

1. Using the Ic vs. Vbe graph, locate the curve whose Vbe(Sat)@Ic/Ib=10. For example:

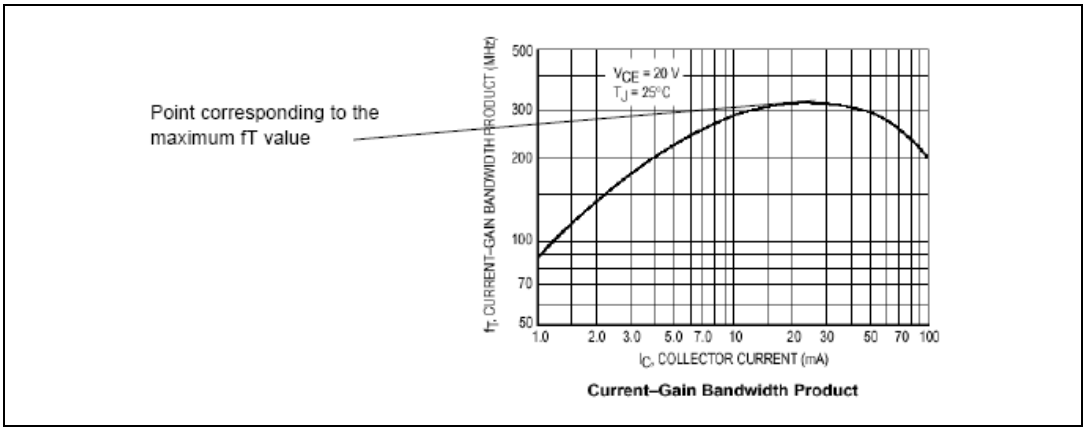

The bottom curve is Vbe-Ic when Vce is saturated and Ic/Ib=10.
Point corresponding to highest value of collector current
Select a point in the high values range of the collector current.
"On" Voltages

2. Find the highest point on the curve. Use the coordinates of this point to enter:
   - **Saturation Collector-Emitter Voltage (Vce2_sat)**
   - **Highest Value of Collector Current**
3. Select a point on the curve in the high values range of the collector current. Use the coordinates of this point to enter:

- **Saturation Collector-Emitter Voltage (Vce1_sat)**
- **Collector Current in the high values range**

Complete the steps below to enter **Current-Gain Bandwidth Product Chart (fT)** data:

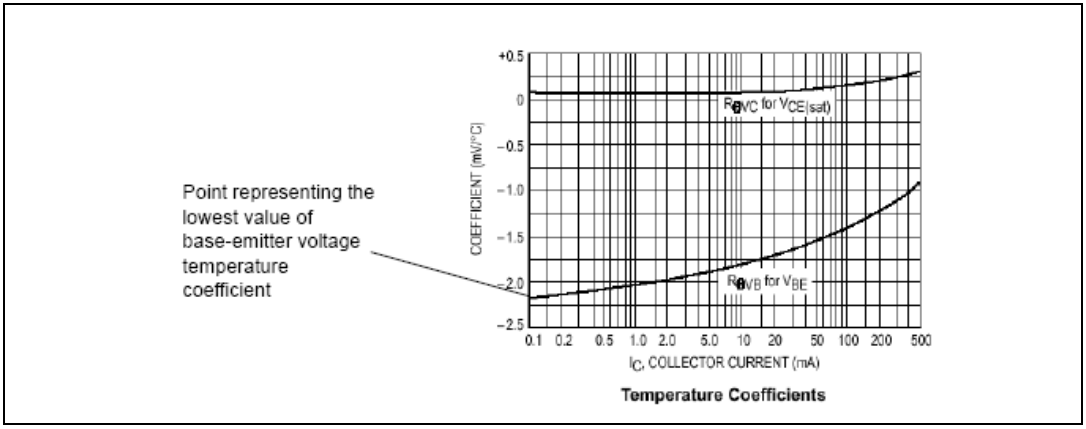
1. In the databook, locate the “Current-Gain Bandwidth Product versus Frequency” graph—for example:



2. Locate the maximum fT value, or the highest point, of the curve. Enter this value in the **Maximum Value of Current-Gain Bandwidth Product** field.

Complete the following steps to enter **Temperature Coefficients Chart** data:

1. In the databook, locate the “Temperature Coefficients” chart—for example:

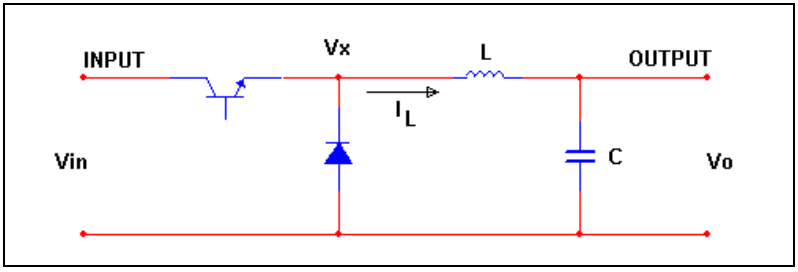


2. On the base-emitter temperature coefficient curve, find the minimum value, or the lowest point, and enter this value in the **Lowest Value of Base-Emitter Voltage Temperature Coefficient** field.

Converters

Buck Converter

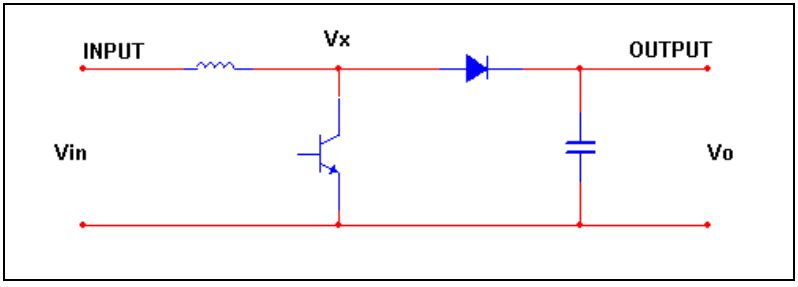
A buck converter is a step-down converter that can be represented by the equivalent circuit shown in the figure below.



Refer to the Buck Converter section for information about this **Model Maker**.

Boost Converter

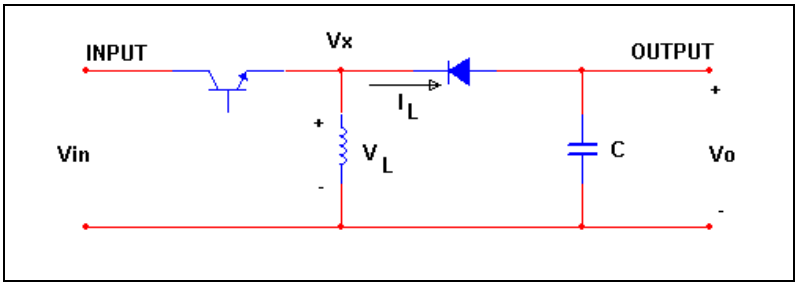
A boost converter is a step-up converter that can be represented by the equivalent circuit shown in the figure below.



Refer to the Boost Converter section for information about this **Model Maker**.

Buck-Boost Converter

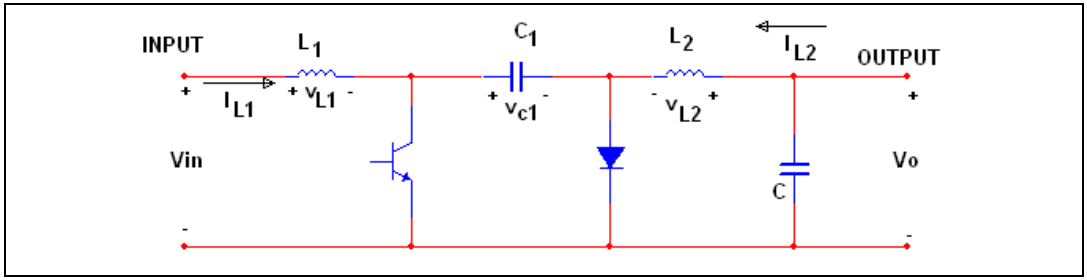
A buck-boost converter is a step-down—step-up converter that can be represented by the equivalent circuit shown in the figure below.



Refer to the Buck Boost Converter section for information about this **Model Maker**.

Cuk Converter

The cuk converter can be represented by the equivalent circuit shown in the figure below.



Refer to the Cuk Converter section for information about this **Model Maker**.

Boost Converter

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Boost Converter** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Boost Converter Model** dialog box appears.

4. Enter desired values in the **Boost Converter Model** dialog box.
5. When all values are entered, click **OK** to complete the model.

Buck Boost Converter

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Buck Boost Converter** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Buck Boost Converter Model** dialog box appears.
4. Enter desired values in the **Buck Boost Converter Model** dialog box.
5. When all values are entered, click **OK** to complete the model.

Buck Converter

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Buck Converter** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Buck Converter Model** dialog box appears.
4. Enter desired values in the **Buck Converter Model** dialog box.
5. When all values are entered, click **OK** to complete the model.

Cuk Converter

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Cuk Converter** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Cuk Converter Model** dialog box appears.
4. Enter desired values in the **Cuk Converter Model** dialog box.
5. When all values are entered, click **OK** to complete the model.

Diode Model Maker

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Diode** and click **Accept**. (Click **Cancel** to return to the Model tab.) The **Diode Model** dialog box appears.
4. Enter values in the **Diode Model** dialog box as described in the following sections.
5. When all values are entered, click **OK** to complete the model.

Note The **Diode Model** dialog box shows preset values for the IN4001 model.

Entering General Data

1. Click the **General** tab.
2. Look up data information for the diode in a databook.

Complete the following steps to enter **Reverse Characteristics** data:

1. In the databook, find the "Maximum Ratings and Electrical Characteristics" table.
2. Based on information in this table, enter the following values:
   - **Maximum repetitive peak reverse voltage (VRRM)**
   - **Maximum DC reverse current at VRRM (IR)**
   - **Typical reverse recovery time (trr)**

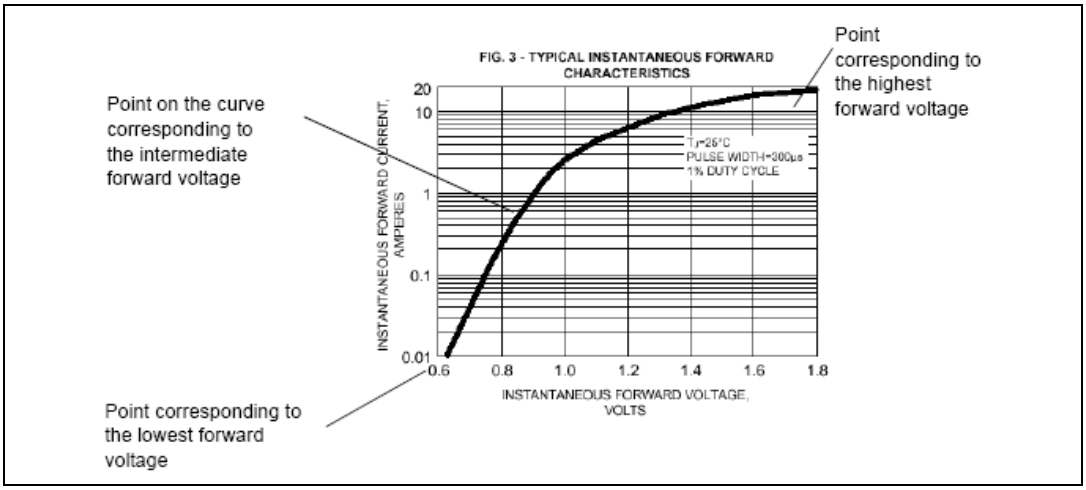
Complete the following steps to enter **Reverse Breakdown** data:

1. In the databook, find the "Reverse Voltage vs. Reverse Current" chart.

   If no data are available, enable **Check if data not available**.
2. On the chart, locate the graph that indicates the ambient temperature of 25° C.
3. Select a point on the graph that represents the mid-point of the horizontal direction, as indicated in the chart.
4. Use the coordinates of this point to enter values for:
   - **Reverse Breakdown Voltage (BV)**
   - **Reverse Breakdown Current (IBV)**

Complete the following steps to enter **Instantaneous Forward Characteristics** data:

1. In the databook, locate the "Typical Instantaneous Forward Characteristics" graph—for example:



2. Find the point of lowest forward voltage, at beginning point of the curve. Use the coordinates of this point to enter values for:
   - **Lowest forward current (IF1)**
   - **Lowest forward voltage (VF1)**
3. Find the point of highest forward voltage, or the end point on the curve. Use the coordinates of this point to enter values for:
   - **Highest forward current (IFM)**
   - **Highest forward voltage (VFM)**
4. Using your eye or a ruler, find the second or intermediate point on the curve which you think best identifies the transition point in the curve.

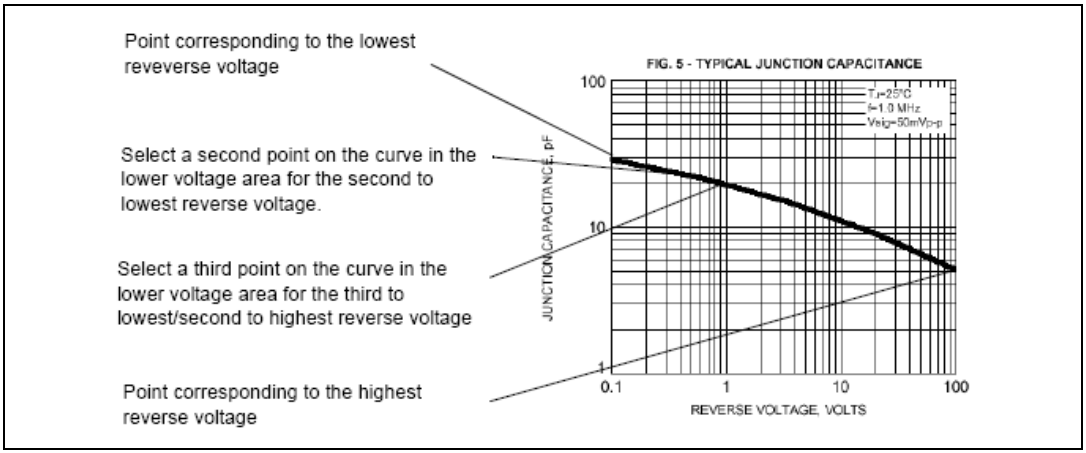
Note Guidelines for selecting the intermediate point vary from one databook to another. If the graph is provided in logarithmic format, which is usually the case, a good way to find this point is to place a ruler along the beginning of the curve in the lower voltage area, which will look like a straight line. Where the curve begins to diverge from your ruler, use this point as your intermediate point. If the graph is provided in linear format, plot the data in logarithmic fashion and follow the ruler procedure.

5. Use the coordinates of this point to enter the values for:
   - **Forward current (IF2)**
   - **Intermediate forward voltage (VF2)**

Entering Capacitance data

Complete the following steps to enter **Junction Capacitances** data:

1. Click the **Junction Capacitance Chart** tab.
2. In the databook, find the "Typical Junction Capacitance" chart—for example:


Point corresponding to the lowest reveverse voltage
Select a second point on the curve in the lower voltage area for the second to lowest reverse voltage.
Select a third point on the curve in the lower voltage area for the third to lowest/second to highest reverse voltage
Point corresponding to the highest reverse voltage
FIG. 5 - TYPICAL JUNCTION CAPACITANCE
Tj=25°C
f=1.0 MHz
Vsig=50mVp-p
JUNCTION CAPACITANCE, pF
100
10
1
0.1
1
10
100
REVERSE VOLTAGE, VOLTS

3. Find the junction capacitance at zero reverse voltage and enter it in the **Junction capacitance at zero reverse voltage (CJO)** field.

   If this information is not given in the databook, enable **Check if data not available**.
4. Find the point of lowest reverse voltage, or the beginning point of the curve. Use the coordinates of this point to enter the values for:
   - **Junction capacitance (CJ1)**
   - **Lowest Reverse Voltage (Vr1)**
5. Find the point of highest reverse voltage, or the end point on the curve) and enter the coordinate values in the **Junction capacitance (CJ4)** and **Highest Reverse Voltage (Vr4)** fields.
6. Select two additional intermediate points on the graph, greater than the lowest reverse voltage but in the lower range of the reverse voltage.
7. Use the coordinate values of the second point to enter:

- **Junction capacitance (CJ2)**
- **Second to Lowest Reverse Voltage (Vr2)**

8. Use the coordinate values of the third point to enter:
   - **Junction Capacitance (CJ3)**
   - **Second to Highest Reverse Voltage (Vr3)**

Transformers

The following transformer models are available in Multisim:

- *Ideal Transformer (Multiple Winding)*
- *Linear Transformer (Multiple Winding)*
- *Linear Transformer with Neutral Terminal*
- *Two Winding Linear Transformer*
- *Non-linear Transformer (Multiple Winding)*

Ideal Transformer (Multiple Winding)

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Ideal Transformer (Multiple Winding)** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Ideal Transformer Model** dialog box appears.
4. In the **Primary** box, click **Option 1** or **Option 2** to quickly select which **Primary Input Voltage** to use.

   Or

   Enter a value directly into the **Primary Input Voltage** fields.
5. Optionally, increase the number of secondary outputs in the **Secondary Output Number** field. The number of **Output Voltage** fields increase to match the entered number.
6. Click **OK** to complete the model.

Linear Transformer (Multiple Winding)

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Linear Transformer (Multiple Winding)** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Linear Transformer Model** dialog box appears.
4. In the **Primary** box, click **Option 1** or **Option 2** to quickly select which primary properties to use.

   Or

   Enter values directly into the **Primary Input Voltage**, **Magnetizing Impedance**, **Leakage Inductance**, and **Winding Resistance** fields.
5. Optionally, increase the number of secondary outputs in the **Secondary Output Number** field. The number of **Output Voltage**, **Leakage Inductance**, and **Winding Resistance** fields increase to match the entered number.
6. Click **OK** to complete the model.

Linear Transformer with Neutral Terminal

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Linear Transformer with Neutral Terminal** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Linear Transformer with Central Terminal in Secondary Side** dialog box appears.
4. Enter the desired values in the dialog box.
5. Click **OK** to complete the model.

Two Winding Linear Transformer

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.

3. From the **Model Maker List**, select **Two Winding Linear Transformer** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Two Winding Linear Transformer Model** dialog box appears.
4. Enter desired values in the dialog box.
5. Click **OK** to complete the model.

Non-linear Transformer (Multiple Winding)

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker List**, select **Non-Linear Transformer (Multiple Winding)** and click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Non-Linear Transformer Model** dialog box appears.
4. Enter desired values in the **Non-Linear Transformer Model** dialog box as described in the following sections.
5. Click **OK** to complete the model.

Entering General Data

1. Click on the **General** tab.
2. In the **Primary** box, click **Option 1** or **Option 2** to quickly select which primary properties to use.

   Or

   Enter values directly into the **Primary Input Voltage**, **Leakage Inductance**, and **Winding Resistance** fields.
3. Optionally, increase the number of secondary outputs in the **Secondary Output Number** field. The number of **Output Voltage**, **Leakage Inductance**, and **Winding Resistance** fields increase to match the entered number.
4. Enter the desired parameters for each of the secondary outputs in the **Output Voltage**, **Leakage Inductance** and **Winding Resistance** fields.
5. Click **OK** to complete the model.

Entering Advanced Data

1. Click on the **Advanced** tab.
2. Enter the **Primary Turns**, **Core Length** and **Core Area** parameters as required.
3. Use one of the pre-set sample B-H curves (**Sample 1**, **Sample 2** or **Sample 3**), or enter your own **B-H Curve** parameters.
4. Click **OK**.

MOSFET (Field Effect Transistor) Model Maker

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker** list, select **MOSFET** and, to continue, click **Accept**. (Click **Cancel** to return to the Model tab.) The **MOSFET Model** dialog box appears.
4. Enter values in the **MOSFET Model** dialog box as described in the following sections.
5. Click **OK** to complete the model.

Note The **MOSFET Model** dialog box shows preset values for the BSS83 model.

Entering General and Output Characteristics Data

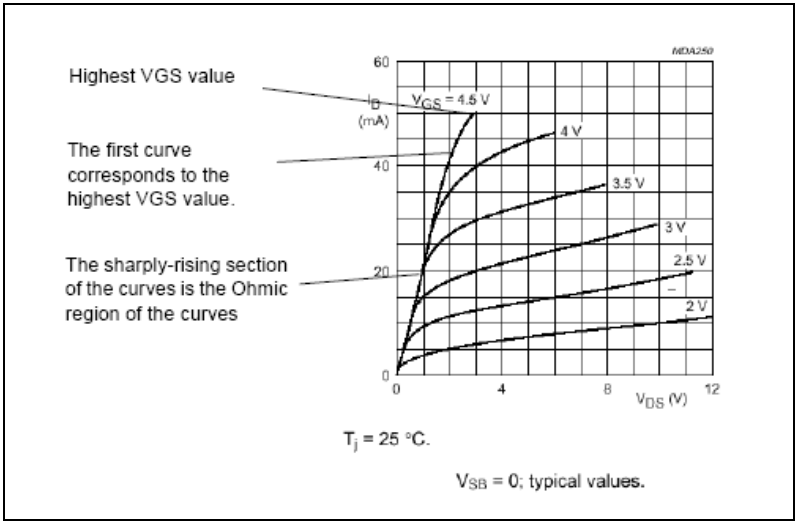
1. Click the **General** tab.
2. Look up data information for the MOSFET in a databook.

Complete the following steps to enter **General** data:

1. Enter the **Channel Type of MOSFET**. This is the title of the datasheet and is found at the top of the datasheet.
2. Find the “Ratings” table for the MOSFET.
3. From the data given in the table, enter the **Max Drain Current**.

Complete the following steps to enter **Output Characteristics in Ohmic Region** data:

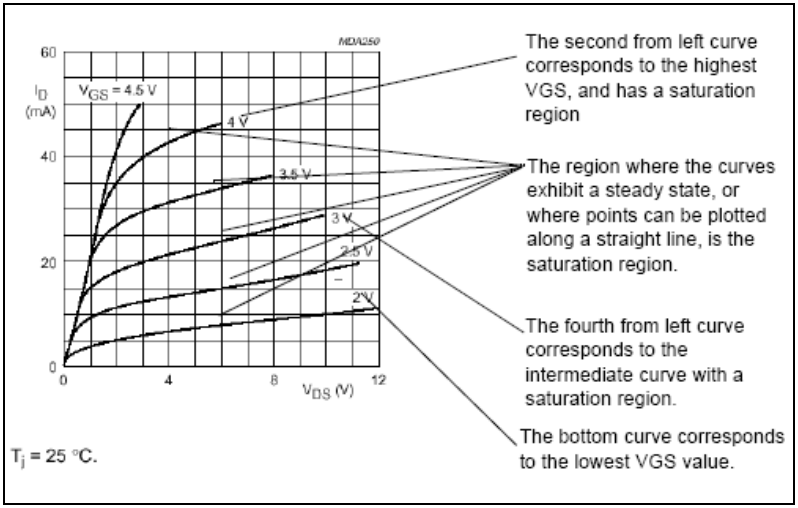
1. From the MOSFET data information, find the Id vs. Vds graph—for example:



2. Locate the curve with the highest Vgs. Enter this Vgs value in the **Vgs for the curve (Vgs_ohmic)**.
3. Locate a point in the ohmic region of the same curve.
4. Enter the Id value of this point in the **Drain Current (Ids_Ohmic)** field.
5. Enter the Vds value of this point in the **Vds when drain current is Ids_Ohmic (Vds_Ohmic)** field.

Complete the following steps to enter other output characteristics:

1. Using the same graph as above, locate the saturation region of the curves. The saturation region is the steady state situation of the curves where points along the curve fit on a straight line. (The curve corresponding to the highest VGS does not have a saturation region.) For example:


MDA250
60
I_D (mA)
V_{GS} = 4.5 V
4 V
3.5 V
3 V
2.5 V
2 V
40
20
0
0
4
8
12
V_{DS} (V)
T_j = 25 °C.
The second from left curve corresponds to the highest VGS, and has a saturation region
The region where the curves exhibit a steady state, or where points can be plotted along a straight line, is the saturation region.
The fourth from left curve corresponds to the intermediate curve with a saturation region.
The bottom curve corresponds to the lowest VGS value.

2. To enter data for the **Ids_Vds curve 1 (for lowest Vgs)** fields, find the curve with the lowest Vgs value.
3. Enter the Vgs value of this curve in the **Vgs for this curve (Vgs_f0)** field.
4. Using your eye or a ruler, locate the beginning and end points of the saturation region, or the area where the points fit along a straight line, for this curve.
5. Use the coordinates of the beginning point to enter:
   - **Drain Current (Ids_f00)**
   - **Vds_f00 (lower Vds)**
6. Use the coordinates of the end point to enter:
   - **Drain Current (Ids_f01)**
   - **Vds_f01 (higher Vds)**
7. To enter data for the **Ids_Vds curve 3 (for highest Vgs)** fields, find the curve with the highest Vgs value, but which still has a saturation region. (This excludes the topmost curve of the Id-Vds graph.)
8. Repeat steps 3 through 7 to enter values for:
   - **Vgs for this curve (Vgs_f2)**
   - **Drain Current**
   - **Vds_f20 (lower Vds)**
   - **Drain Current (Ids_f21)**
   - **Vds_f21 (higher Vds)**

9. To enter data for the **Ids_Vds curve 2 (for intermediate Vgs)** fields, find the curve in the middle point between the curves corresponding to the lowest Vgs and the highest VGS with a saturation region.
10. Repeat steps 3 through 7 to enter values for:
    - **Vgs for this curve (Vgs_f1)**
    - **Drain Current (Ids_f10)**
    - **Vds_f01 (lower Vds)**
    - **Drain Current (Ids_f11)**
    - **Vds_f11 (higher Vds)**

Entering Transfer Characteristics data

Complete the following steps to select Transfer Characteristics options:

1. Click the **Transfer Characteristics - Option A, B** tab.
2. In the databook, locate the Id vs. Vgs graph. Depending on the available data, under **Vds for Id-Vgs curve** and **Multi Id-Vgs curve**, enable the appropriate options.

Note If the graph contains more than one Vsb curve, it implies that source and bulk (substrate) are not connected together.

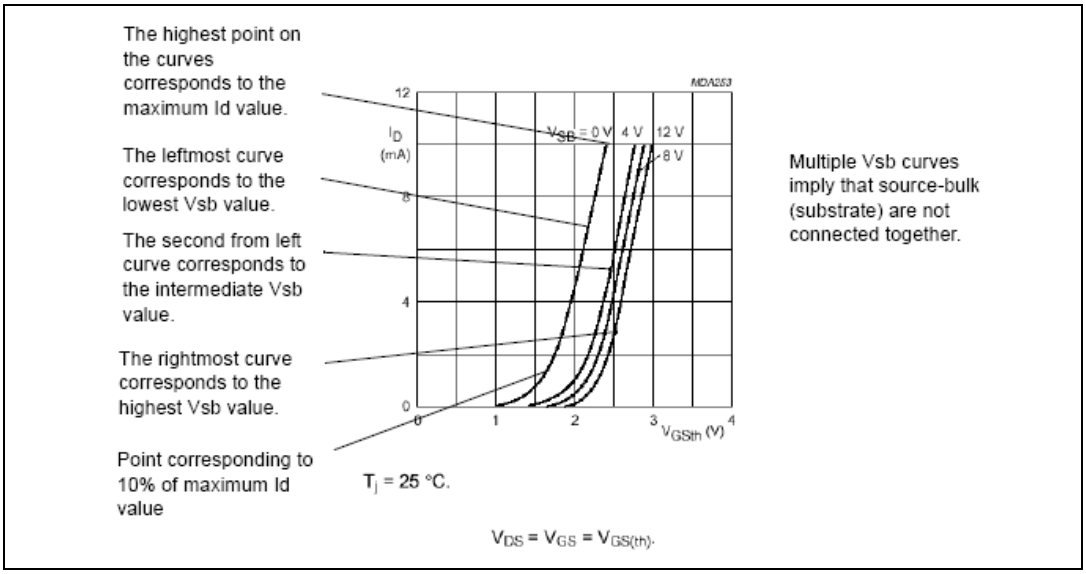
   If the latter option is not enabled, you will be prompted to enter data in the **Option A** or **Option B** fields on the same dialog box.

   If the latter option is enabled (as it is in our example), the dialog box will prompt you to **Go to Option C (Next Page)**.

3. To go to Option C, click the **Transfer Characteristics - Option C** tab.

Complete the following steps to enter **Transfer Characteristics (Drain Current vs. Gate-to-Source Voltage)** data for all three options (Option A, B, or C):

1. Look at the data in the Ids vs. Vgs graph—for example:



2. If you are using Option A, proceed to step 4.

   If you are using Option B, proceed to step 3.

   If you are using Option C, to enter data in the **Ids-Vgs Curve for lowest Vsb** fields, locate the curve with the lowest Vsb.

3. If you are using Option B, enter the Vds value in the **Vds for Ids-Vgs curve** field.

   If you are using Option C, enter the Vsb value in the **Vsb for this curve (Vsb_C1)** field.

4. Find the maximum Id, or the highest point of the curve. Use the coordinates for this point to enter:

   for Option A:

   - **Drain Current (Ids_A1)**
   - **Vgs_A1 (lower Vgs)**

   for Option B:

   - **Drain Current Ids_B1**
   - **Vgs_B1 (lower Vgs)**

for Option C:

- **Drain Current (Ids_C11)**
- **Vgs-C11 (lower Vgs)**

5. Find the point on the curve which corresponds to 10% of the maximum Id on the same curve. Use the coordinates of this point to enter:

   for Option A:

   - **Drain Current (Ids_A2)**
   - **Vgs_A2 (higher Vgs)**

   for Option B:

   - **Drain Current (Ids_B2)**
   - **Vgs_B2 (higher Vgs)**

   for Option C:

   - **Drain Current (Ids_C12)**
   - **Vgs_C12 (higher Vgs)**

6. To complete the Option C dialog box, to enter data in the **Ids-Vgs Curve for highest Vsb**, find the curve with the highest Vsb value, and repeat steps 3 through 5 above to enter data for:

   - **Vbs for this curve (Vsb_C3)**
   - **Drain Current (Ids_C31)**
   - **Vgs_C21 (lower Vgs)**
   - **Drain Current (Ids_C32)**
   - **Vgs_C22 (highest Vgs)**

7. To enter data in the **Ids-Vgs Curve for Intermediate Vsb** fields, select the curve corresponding to a Vsb value in between the highest and lowest Vsb. Repeat steps 3 through 5 above to enter data for:

   - **Vsb for this curve (Vsb_C2)**
   - **Drain Current (Ids_C21)**
   - **Vgs_C21 (lower Vgs)**
   - **Drain Current (Ids_C22)**
   - **Vgs_C22 (highest Vgs)**

Entering Capacitances Data

Complete the following steps to enter **Conditions**:

1. Click the **Junction Capacitance Chart** tab.
2. Determine whether the bulk and source of the model are connected, and select the appropriate answer beside **Bulk and Source connected**.

   The substrate condition can be determined by two means. The first is to check the schematic of the device where the internal connections of the MOS transistor are shown.

Note The second is to check the Id-Vgs graph. If the graph contains more than one Vsb curve, it suggests that source-bulk (substrate) are not connected together.

3. In the databook, locate the "Capacitances vs. Drain-to-Source Voltage" chart. If it is available, you may enter data in the **Capacitances vs. Drain-to-Source Voltage** fields. If it is not available, enable **Coss-Vds and Crss-Vds curve NOT available**, and use the datasheet to enter capacitances.

Complete the following steps to enter **Capacitance values from datasheet**:

1. In the databook, find the "Characteristics" table.
2. From the table, enter data for:
   - **Feedback capacitance**
   - **Input capacitance**
   - **Output capacitance**
   - **Drain-Substrate Voltage**

Operational Amplifier Model Maker

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit.** The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker** list, select **Operational Amplifier** and, to continue, click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Operational Amplifier Model** dialog box appears.
4. Enter values on the **Operational Amplifier Model** dialog box as described in the following sections.
5. When all values are entered, click **OK** to complete the model.

Note The **Operational Amplifier Model** dialog box shows preset values for the uA741 model.

Entering General and Input Data

1. Click the **General, Input** tab.
2. Look up data information for the operational amplifier in a databook.

Complete the following steps to enter **General** data:

1. In the **Transistor Type** field, select the type of transistor used in the input stage. This can be determined by looking at the schematic of the internal structure of the op-amp.

Note This information is optional, as the op-amp model can be based on any type of input transistor. If the type of the input transistor is not important, select **Don't Care**.

Complete the following steps to enter input data:

1. In the databook, find the two tables labeled "Electrical Characteristics at specified free-air temperature".
2. Use the data from these tables to enter:
   - **Input Capacitance (Ci)**
   - **Input Offset Current (I1o)**
   - **Input Bias Current ((I1b)**
   - **Input Offset Voltage (V1o)**
   - **Common-mode Input Resistance (Rcm)**
   - **Differential-mode Input Resistance (R1)**
   - **Common-mode Rejection Ratio (CMRR)**
   - **Voltage Gain Avd**

Note The Common-Mode input resistance is usually very high. If its value is not available, choose 2 Gohm as the default.

Note While the typical value for the Common-Mode Rejection Ratio (CMRR) is provided in the "Electrical Characteristics" table, its variations with frequency are also provided in a chart called "Common-Mode Rejection Rate Vs. Frequency". If you use this chart, use the CMRR value for the lowest frequency possible.

Note While the typical value for Large Signal Differential Voltage Amplification (Avd) is provided in the "Electrical Characteristics" table, you can also find it in a chart called

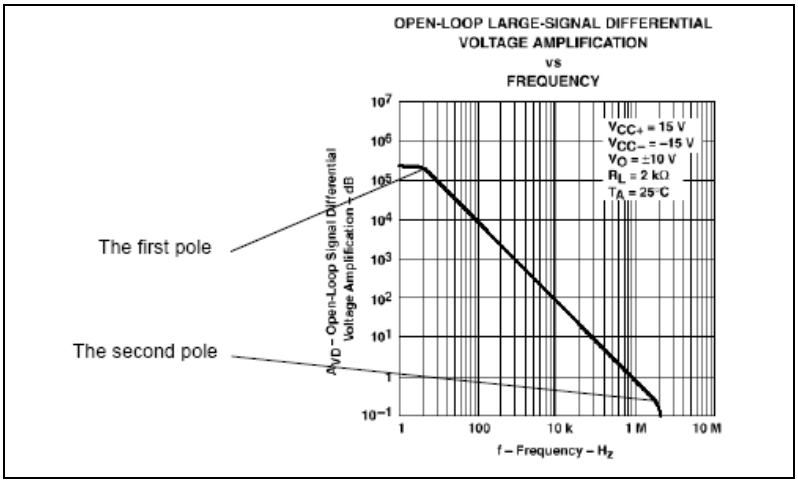
"Open-Loop Large Signal Differential Voltage". If you use this chart, use the Avd value at the lowest frequency.

Note Databooks provide Avd gain in either dB or V/mV. If the value is provided in V/mV, you can still enter the data in dB. However, you should convert the numerical values: *value in dB = 20 * log[1000 * (value in V/mv)]*

Entering Poles and Zero Data

Complete the following steps to enter **Gain-frequency curve poles and Zero** data:

1. Click the **Gain frequency curve poles and Zero** tab.
2. In the databook, locate the "Avd-Open-Loop Single Differential Voltage Amplification vs. Frequency" chart—for example:



3. Find the first pole on the curve, or the point on the curve where the first horizontal line transitions into a slope. Enter the frequency value for this point in the **Pole 1 frequency (fr1)** field.
4. Find the second pole on the curve, or the point where the slope transitions into a sharper slope. Enter the frequency value for this point in the **Pole 2 frequency (fr2)** field.

To enter **High frequency pole and zero** data, find higher frequency poles using the curve mentioned above, web sites or books. If these pieces of information are not available, enable **Not Available**.

Entering Output Data

1. Click the **Output** tab.
2. In the databook, locate the "Operating Characteristics" table.
3. Use the data from this table to enter:
   - **Slew Rate (SR+) (non-inverting mode)**
   - **Slew Rate (SR-) (inverting mode)**

Note Databooks may provide only one value for both inverted and non-inverted slew rates.

4. Refer to the "Electrical Characteristics" tables mentioned in the previous section. Use the data from these tables to enter:
   - **Output Resistance (Ro)**
   - **Maximum Source Current (Isr)**
   - **Maximum Sink Current (Isn)**

Note Databooks normally provide the short circuit output current. This is the maximum value of the output current which the output node can provide if it is connected to the negative power supply, or can accept if it is shorted to the positive power supply. You should enter its value regardless of its sign.

Silicon Controlled Rectifier Model Maker

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker** list, select **SCR** and, to continue, click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **SCR Model** dialog box appears.
4. Enter values on the **SCR Model** dialog box as described in the following sections.
5. When all values are entered, click **OK** to complete the model.

Note The **SCR Model** dialog box shows preset values for the 2N6504 SCR.

Entering Electrical and Maximum Forward Voltage Data

1. Click the **Electrical Characteristics, Max Forward Voltage** tab.
2. Look up data information for the SCR in a databook.

Complete the following steps to enter **Electrical Characteristics** data:

1. Locate the “Electrical Characteristics” table—for example:

Use this information to enter data in the Electrical Characteristics fields under the Electrical Data, Max Forward Voltage tab.

Use this information to enter data in the Electrical Characteristics fields of under the Time Data, Max Ratings tab.

ELECTRICAL CHARACTERISTICS (T_C = 25°C unless otherwise noted.)

Characteristic		Symbol	Min	Typ	Max	Unit
*Peak Forward or Reverse Blocking Current (V_{AK} = Rated V_{DRM} or V_{RRM}, Gate Open)	T_J = 25°C T_J = 125°C	I_{DRM}, I_{RRM}	— —	— —	10 2	μA mA
*Forward “On” Voltage(1) (I_{TM} = 50 A)		V_{TM}	—	—	1.8	Volts
*Gate Trigger Current (Continuous dc) (Anode Voltage = 12 Vdc, R_L = 100 Ohms)	T_C = 25°C T_C = –40°C	I_{GT}	— —	— 25	40 75	mA
*Gate Trigger Voltage (Continuous dc) (Anode Voltage = 12 Vdc, R_L = 100 Ohms, T_C = –40°C)		V_{GT}	—	1	1.5	Volts
Gate Non-Trigger Voltage (Anode Voltage = Rated V_{DRM}, R_L = 100 Ohms, T_J = 125°C)		V_{GD}	0.2	—	—	Volts
*Holding Current (Anode Voltage = 12 Vdc, T_C = –40°C)		I_H	—	35	40	mA
*Turn-On Time (I_{TM} = 25 A, I_{GT} = 50 mAdc)		t_{gt}	—	1.5	2	μs
Turn-Off Time (V_{DRM} = rated voltage) (I_{TM} = 25 A, I_R = 25 A) (I_{TM} = 25 A, I_R = 25 A, T_J = 125°C)		t_q	— —	15 35	— —	μs
Critical Rate of Rise of Off-State Voltage (Gate Open, Rated V_{DRM}, Exponential Waveform)		dv/dt	—	50	—	V/μs

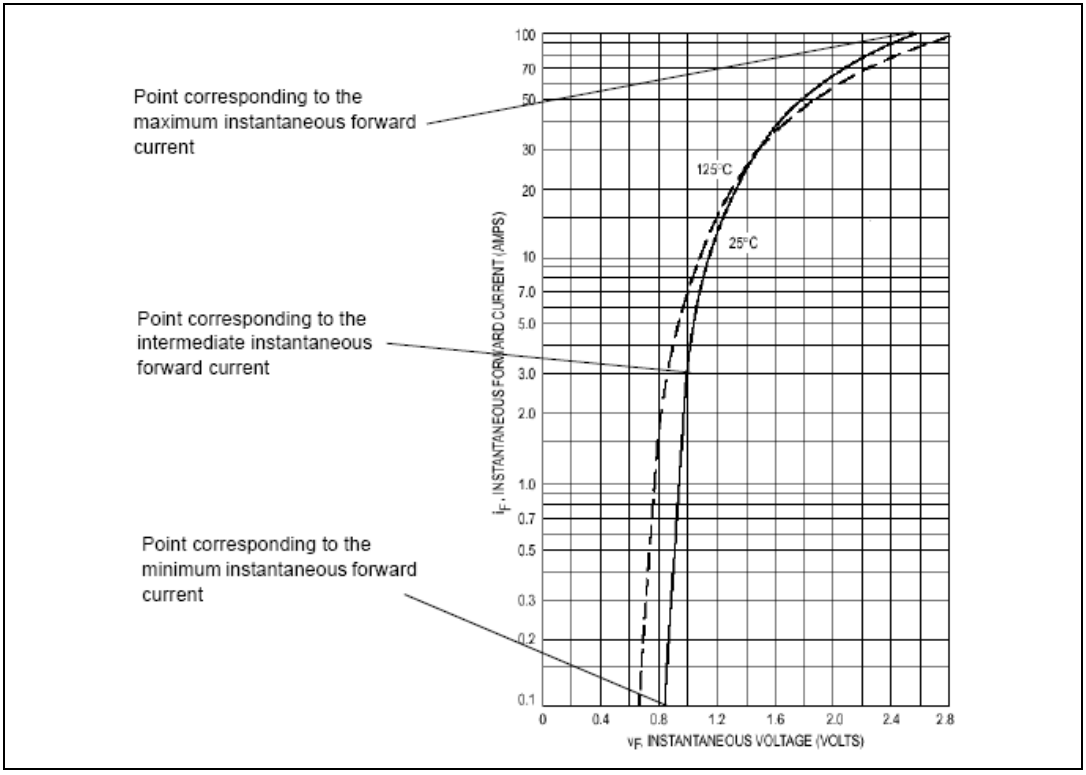
*Indicates JEDEC Registered Data.

1. Pulse Test: Pulse Width ≤ 300 μs, Duty Cycle ≤ 2%.

2. Based on the data provided in this table, enter:
   - **Holding Current**
   - **Gate Trigger Current**
   - **Gate Trigger Voltage**
   - **Peak Forward Blocking Current**
   - **Critical Rate of Rise of Off-State Voltage**
3. In the databook, locate the “Maximum Ratings” table.
4. Based on the data in the table, enter the value of the **Peak Forward Blocking Voltage** field.
5. Multiply this value by 1.05 and enter the value in the **1.05 Peak Reverse Blocking Voltage** field.

Complete the following steps to enter **Maximum Forward Voltage Chart** data:

1. In the databook, locate the “Instantaneous Forward Current vs. Instantaneous Voltage” graph, and find the If-Vf curve at 25°. For example:



2. On the curve, find the point at the minimum If, or the beginning point of the curve. Use the coordinates of this point to enter:
   - **Instantaneous Forward Current**
   - **Minimum value of instantaneous voltage**
3. Find the point at the maximum If, or the end point of the curve. use the coordinates of this point to enter:
   - **Instantaneous Forward Current**
   - **Maximum value of instantaneous voltage**

4. Locate an intermediate point on the curve corresponding to the transition point. Since the graph is provided in logarithmic format, you can do this by using a ruler to draw a line starting at the beginning point and following the straight line of the curve in the lower voltage area. Where the curve begins to diverge from your ruler, use this point as your intermediate point. Use the coordinates of this point to enter:
   - **Instantaneous Forward Current**
   - **Intermediate value of instantaneous voltage**

Entering Time Data and Maximum Ratings Data

1. Click the **Time Data, Max Ratings** tab.

To enter **Electrical Characteristics** data, refer to the "Electrical Characteristics" table mentioned in the previous section, and enter data in the **Turn-On Time** and **Turn-Off Time** fields.

Complete the following steps to enter **Maximum Ratings Chart** data:

1. Refer to the "Maximum Ratings" table mentioned in the previous section.
2. Find the Forward Current and enter this value in the **Forward Current** field.
3. For the **Reverse Current** field, find the reverse current (IRC) when the device is in off-state and enter this value, or, if this value is not provided, enter 0.
4. For the **Reverse Voltage** field, find the reverse voltage (VRC) when the device is in off-state or, if this value is not provided, enter 0.
5. For the **Identifier** field, enter 1 if Reverse Current and Reverse Voltage values are available, or 0 if they are unavailable.
6. For the first Parameter related to "off-state" field, enter 0 if the Reverse Current and Reverse Voltage values are available, or, if they are not provided, enter the Forward Current value.
7. For the second **Parameter related to "off-state"** field, enter 0 if the Reverse Current and Reverse Voltage values are available, or, if they are not provided, enter the Peak Reverse Blocking Voltage value.

Zener Model Maker

1. From the **Model** tab of the **Component Properties** dialog box, click **Add/Edit**. The **Select a Model** dialog box appears.
2. Click **Start Model Maker**. The **Select Model Maker** dialog box appears.
3. From the **Model Maker** list, select **Zener** and, to continue, click **Accept**. (Click **Cancel** to return to the **Model** tab.) The **Zener Model** dialog box appears.
4. Enter values on the **Zener Model** dialog box as described in the following sections.
5. When all values are entered, click **OK** to complete the model.

Note The **Zener Model** dialog box shows preset values for the IN4728A model.

Entering Electrical Characteristics Data

1. Click the **Electrical Characteristics** tab.
2. Look up data information for the Zener diode in a databook.
3. From the databook, locate the "Electrical Characteristics" table—for example:

Use the information from this table to enter data in the Reverse Characteristics fields.

*ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted) V_F = 1.2 V Max, I_F = 200 mA for all types.

JEDEC Type No. (Note 1)	Nominal Zener Voltage V_Z @ I_{ZT} Volts (Notes 2 and 3)	Test Current I_{ZT} mA	Maximum Zener Impedance (Note 4)			Leakage Current		Surge Current @ T_A = 25°C I_r – mA (Note 5)
			Z_{ZT} @ I_{ZT} Ohms	Z_{ZK} @ I_{ZK} Ohms	I_{ZK} mA	I_R µA Max	V_R Volts	
1N4728A	3.3	76	10	400	1	100	1	1380
1N4729A	3.6	69	10	400	1	100	1	1260
1N4730A	3.9	64	9	400	1	50	1	1190
1N4731A	4.3	58	9	400	1	10	1	1070
1N4732A	4.7	53	8	500	1	10	1	970
1N4733A	*5.1*	*49*	*7*	*550*	*1*	*10*	*1*	*890*
1N4734A	*5.6*	*45*	*5*	*600*	*1*	*10*	*2*	*810*
1N4735A	*6.2*	*41*	*2*	*700*	*1*	*10*	*3*	*730*
1N4736A	*6.8*	*37*	*3.5*	*700*	*1*	*10*	*4*	*660*
1N4737A	7.5	34	4	700	0.5	10	5	605
1N4738A	*8.2*	*31*	*4.5*	*700*	*0.5*	*10*	*6*	*550*

To enter **Reverse Characteristics** data, use the information from the table for the following fields:

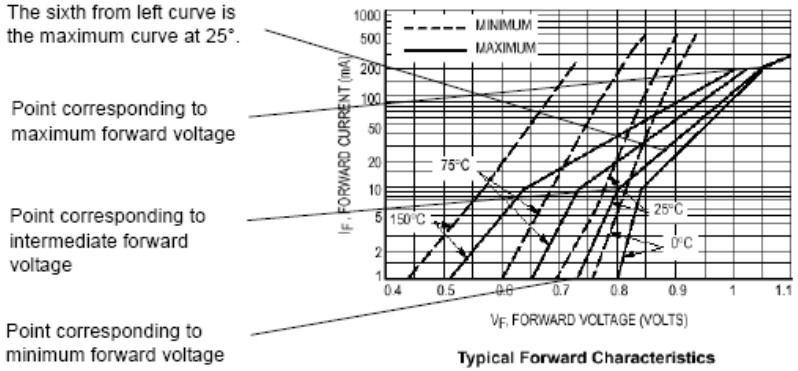
- **Nominal Zener Voltage (Vzt)**
- **Zener Test Current (Izt)**
- **Nominal Zener Impedance at Zzt and Izt (Zzt)**
- **Leakage Current (Ir)**

- **Reverse Test Voltage (Vr)**
- **Zener Impedance near Breakdown Knee (Zzk)**
- **Zener Current near Breakdown Knee (Izk)**

Note In the example, the databook only provides the maximum Zener impedance. To find a typical value for Zzk, use 0.75 times the maximum value of Zzk. To find the typical value of Zzt, you can use the Zz-Iz graph. Find or estimate a curve at the nominal zener voltage given in the table, and choose the point which corresponds to the test current given in the table. Use the Zz coordinate of this point to enter as the typical value.

Complete the following steps to enter **Forward Characteristics** data:

1. In the databook, locate the If-Vf graph, and find the maximum curve at 25°—for example:



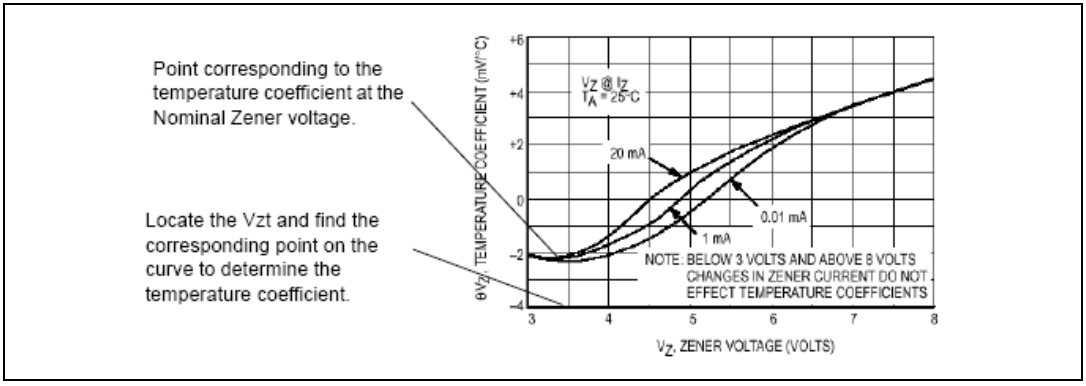
Typical Forward Characteristics

2. Find the point on the curve with the lowest forward voltage, or the beginning point. Use the coordinates of this point to enter:
   - **Forward Current (If1)**
   - **Lowest Forward Voltage (Vf1)**
3. Find the knee point on the curve, or the point where the slope changes drastically. Use the coordinates of this point to enter:
   - **Forward Current (If2)**
   - **Intermediate Forward Voltage (Vf2)**
4. Find the point of maximum forward voltage, or the highest point on the curve. Use the coordinates of this point to enter:
   - **Highest Forward Current (Ifm)**
   - **Highest Forward Voltage (Vfm)**

Entering Temperature Coefficient and Capacitance Data

Complete the following steps to enter **Temperature Coefficient** data:

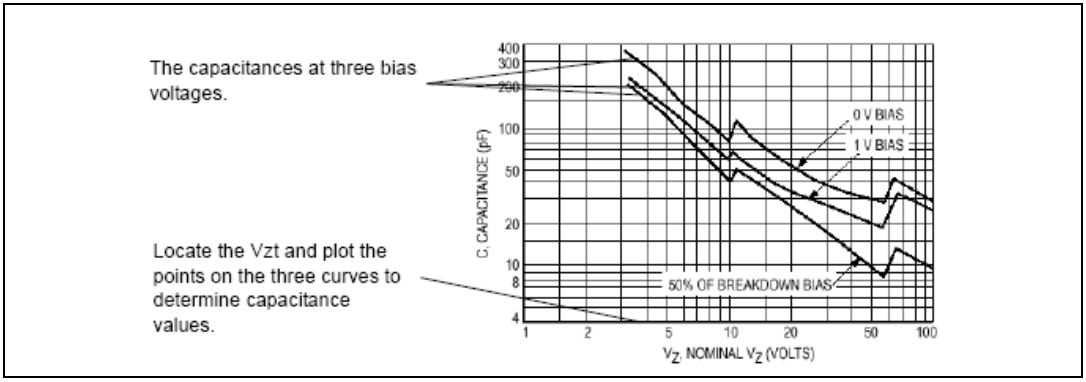
1. Click the **Temperature Coefficient, Capacitance** tab.
2. From the databook, find the Temperature Coefficient versus Zener Voltage graph—for example:


Point corresponding to the temperature coefficient at the Nominal Zener voltage.
Locate the Vzt and find the corresponding point on the curve to determine the temperature coefficient.
θVZ, TEMPERATURE COEFFICIENT (mV/°C)
VZ @ IZ TA = 25°C
20 mA
1 mA
0.01 mA
NOTE: BELOW 3 VOLTS AND ABOVE 8 VOLTS CHANGES IN ZENER CURRENT DO NOT EFFECT TEMPERATURE COEFFICIENTS
VZ, ZENER VOLTAGE (VOLTS)

3. Locate the curve corresponding to the model's test current as given in the "Electrical Characteristics" table. (If it is not on the graph, estimate its placement.)
4. Find the point corresponding to the model's Vz, as provided in the "Electrical Characteristics" table. Enter the Temperature Coefficient for this point in the **Temperature Coefficient at Zener Nominal Voltage (THETA_vz)** field.

Complete the following steps to enter **Capacitance vs. Bias Voltage** data:

1. In the databook, locate the "Capacitance versus Nominal Vz" graph—for example:


The capacitances at three bias voltages.
Locate the Vzt and plot the points on the three curves to determine capacitance values.
C, CAPACITANCE (pF)
0 V BIAS
1 V BIAS
50% OF BREAKDOWN BIAS
VZ, NOMINAL VZ (VOLTS)

2. On each of the three curves in the graph, locate the point corresponding to the Vzt provided in the "Electrical Characteristics" table.
3. For the curve at zero bias voltage, use this point to enter the capacitance value in the **Capacitance at 0 Bias Voltage (CJ1)** field.
4. For the intermediate curve, enter its bias voltage in the **Intermediate Bias Voltage** field and enter the capacitance value for the point you have marked in the **Capacitance (CJ2)** field.
5. For the curve with the highest voltage, enter its bias voltage in the **Highest Voltage** field. (In our example, this value is 50% of the nominal Zener voltage (Vzt) of the model, as provided in the "Electrical Characteristics" table.) Enter the capacitance value for the point you have marked in the **Capacitance (CJM)** field.

Creating a Model Using Code Modeling

This section explains how to model a component using a high-level, industry-standard programming language: C. The component can then be added to the Multisim database. To use code modeling you must have a C compiler such as Microsoft Visual C++, Version 4.1 or greater, and be familiar with programming and compiling C code. This section is not designed for Multisim users without programming exposure.

Multisim has built-in models for most types of devices, but it is impossible to provide models for every possible device. The behavior of some devices may be extremely difficult to model as groups of SPICE components, but may be easier to describe in terms of high-level, behavioral equations. As a result, the behavior of these devices can be modeled using code modeling.

Note Refer to the *Editing a Component's Model* section for other useful information.

What is Code Modeling?

Code modeling is the behavioral modeling of devices whose governing equations are known.

Note This section serves as a basic guide to code modeling and includes helpful examples. However, code modeling is a complex process, so be aware that you need time and practice to gain proficiency.

A code model consists of a set of interface definitions and a C function implementation describing the device's behavior. The naming and location of these files is important. The model is created by combining two files (`Ifspec.ifs` and `Cfunc.mod`). The resulting file, which is given the same name as the folder containing its source files, is placed in the `codemodl` folder.

Creating a Code Model

Refer to the *GAINTEST Example* section for a specific example of compiling and using a code mode.

For best results, do all of the following steps from a DOS command window.

Complete the following steps to create a code model:

1. Set up your environment variables for Microsoft Visual C++ by running VcVars32.bat (installed, by default, in the `C:/Program Files/Microsoft Visual Studio/Vc98/Bin` folder).
2. Navigate to `C:\Documents and Settings\All Users\Shared Documents\National Instruments\Circuit Design Suite 10.1\codemodl`.

Note For systems using the Windows Vista operating system, the path is `C:\users\Public\Documents\National Instruments\Circuit Design Suite 10.1\codemodl`.

Within this directory is a single subdirectory called `USER`. You should create new subdirectories inside of `USER`, one for each of your code models. The name of the subdirectory will end up being the name of the code model executable.

3. Create a new interface file inside your model's subdirectory called `Ifspec.ifs` to set up your model's basic definitions and interface (i/o ports and parameters).
4. Create a new implementation file inside your model's subdirectory called `Cfunc.mod`. This file contains the actual code model.

   It is critical that the `Cfunc.mod` file include a list of all models in the file in the following format:

```
SPICEdev * FAR DynDEVices[] = {
    &<function_name>_info
};
```

Where <function_name> is the "C_Function_Name" defined in your Ifspec.ifs file. This is a special Multisim requirement, so this line may need to be added to code models used from other sources. It is recommended that you use a function name beginning with "cm_".

5. To compile the files into a form that can be used by Multisim, go to the codemodl\USER folder and execute the following command:

   ```
   MAKE_DLL <subdirectory>
   ```

   Where <subdirectory> is the directory where you have put your code model files.

6. If the compilation is successful, you will find a new .dll file inside the subdirectory, which is the executable code model. You *must* copy or move this .dll file into the main codemodl directory in order for Multisim to find it.

7. **Important:** If Multisim is running, you must exit and restart it in order to access new code models. Multisim only loads user code models on startup.

8. In order to use your code model in Multisim, choose a model in the normal way in the component wizard, and select "Load from File". Set the file type in the dialog to "**Code Model DLL(*.dll)**", and browse to the codemodl directory if necessary. Select the .dll file you just created.

9. If you have not altered the codemodl directory structure, the model name and spice model name will be automatically populated from the ifspec.ifs file. The format should look as follows:

   ```
   .MODEL <DLL> <name>(<params>)
   ```

   where <DLL> is the .dll name of your model (without the .dll extension), <name> is the "Spice_Model_Name" from your Ifspec.ifs, and <params> is an optional list of the type "<param_name> = <default value>" separated by spaces (*not* commas), and serves to override any default parameter values defined in Ifspec.ifs. Delete the string "_INSERT_PARAMS" and type in your desired parameters, or leave the parentheses empty if you wish to use all defaults.

GAINTEST Example

The GAINTEST example is included in the Multisim install. This example assumes familiarity with the **Component Wizard**.

1. Set up your environment variables for Microsoft Visual C++ by running `VcVars32.bat` (installed, by default, in the `C:/Program Files/Microsoft Visual Studio/Vc98/Bin` folder).
2. Navigate to `C:\Documents and Settings\All Users\Shared Documents\National Instruments\Circuit Design Suite 10.1\codemodl\USER\GAINTEST`. Notice that it contains only two files: `Ifspec.ifs` and `Cfunc.mod`.

Note For systems using the Windows Vista operating system, the path is `C:\users\Public\Documents\National Instruments\Circuit Design Suite 10.1\codemodl\USER\GAINTEST`.

3. If you wish, you may examine the **`Ifspec.ifs`** file for the `GAINTEST` model. This defines many of the important characteristics of the `GAINTEST` model:
   - The `C_Function_Name (cm_gaintest)`
   - The `Spice_Model_Name (gaintest)`
   - One input port (analog: voltage, differential voltage, current, or differential current).
   - One output port (same options as the input).
4. Three model parameters (`in_offset`, `gain`, and `out_offset`). Note that each parameter is given a plaintext description, a data type, and a default value. All three of them have `Null_Allowed` defined as "yes", so they are all optional parameters. If any of these parameters are omitted when calling the model, the default value will be used.
5. If you wish, you may examine the `Cfunc.mod` file for the `GAINTEST` model. This file contains the actual code model – a simple C language function with the same name defined as `C_Function_Name` in the `Ifspec.ifs` file. Note use of XSpice keywords such as `INPUT`, `OUTPUT`, `PARAM`, `PARTIAL`, and `AC_GAIN`. Note also the use of the model parameters defined in the `Ifspec.ifs` file.
6. Note that the `GAINTEST` **`Cfunc.mod`** file includes the following:

```
SPICEdev * FAR DynDEVices[] = {
    &cm_gaintest_info
};
```

   as required by Multisim.

7. To compile the GAINTEST model, go to the codemodl\USER folder and execute the following command:

   ```
   MAKE_DLL GAINTEST
   ```

   After successful completion, note that two new files have been created in the GAINTEST directory: Gaintest.c and Gaintest.dll:

   - The Gaintest.c file contains all the information from the Ifspec.ifs and Cfunc.mod files, expanded into a format understood by the Microsoft Visual C compiler and the Multisim implementation of XSpice. If you examine this file, you'll notice that all of the XSpice keywords have been replaced with fairly complex structure and array references.
   - The Gaintest.dll file contains the compiled code model, ready for execution.
8. Move this Gaintest.dll file into the main codemodl directory so that Multisim may find it.
9. If Multisim is running, you must exit and restart it so that it may load the new GAINTEST model. Multisim only recognizes new code models on startup.
10. In Multisim, run the component wizard as follows:
    - Create an analog component, simulation only.
    - Give the part two pins, one input and one output.
    - On the "Select Simulation Model" step, click on the "Load from File" button.
    - In the file-selection dialog that appears, change the filter to "Code Model DLL(*.dll)" and browse to the codemodl directory if necessary.
    - Select the Gaintest.dll file and click "Open".
    - The model name should change to "GAINTEST" and the model data to ".MODEL GAINTEST gaintest(_INSERT_PARAMS)"
    - In the "Model Data" section, change _INSERT_PARAMS to **gain=2.0** (this creates a part that multiplies the input voltage by two).
    - Finally, put this model in some logical place in your **User Database**.
11. Create a new circuit and place your new part in series. You may verify that any voltage applied to the part (DC or AC) will be doubled on output. If you wish, you may edit the part's model on the circuit to

adjust the gain parameter, or add in values for the `in_offset` or `out_offset` parameters as well.

The Interface File (Ifspec.ifs)

The interface file sets out, in tables, the names used by the model, the electrical connections to the devices (ports), and the user-defined variables (parameters) that provide finer control over the behavior of the model. These tables are explained in the following sections, with examples given for each.

Refer to the *Example Interface File* section for an example of an interface file. The interface file, along with the implementation file, must be compiled into a DLL to complete the code model.

Name Table

The model name, description text, and C implementation function name are defined in the name table. The model name must be the same as the subdirectory containing the code model files. It is recommended that the model name be eight characters.

The name table has the following syntax:

```
NAME_TABLE:

C_Function_Name:function_name
Spice_ModelName:model_name
Description:    "text"
```

where:

function_name — is a valid C identifier which is the name of the main entry point (function) for the code model. It may or may not be the same as the SPICE model name. To reduce the chance of name conflicts, we recommend you use the prefix "UCM_" for user code model, or use a prefix based on your own initials. The following prefixes are used by the XSPICE simulator core and should not be used for user code models:

A2VERI	D_OPEN_C	EW_RES	POLY
A2VHDL	D_OPEN_E	EW_SCR	POT
A_555	D_OR	EW_SWTCH	PPT
ADC_BRDG	D_OSC	EW_VLT	PWL
ASRC	D_PULLDN	FTE	R_2_V
ASWITCH	D_PULLUP	GAIN	RDELAY
BJT	D_RAM	HLP	RES
BSIM	D_ROM	YST	RGAIN
CAP	D_SERIALP	ICM	S_XFER
CCCS	RT	IDN	SCR
CCVS	D_SOURCE	ILIMIT	SINE
CKT	D_SRFF	IND	SLEW
CLIMIT	D_SRLATC	INDUCTOR	SMP
CM	D_STATE	INP	SQUARE
CMETER	D_TFF	INT	SRC_LVM
CORE	D_TRISTA	IPC	SRC_LVM_
CP	D_VERI	ISRC	UR
CSW	D_VERILOG	JFET	SRC_TDM
D_2_R	D_VHDL	LCOUPLE	SRC_TDM_
D_AND	D_WGEN	LIMIT	UR
D_BUFFER	D_XNOR	LMETER	SUMMER
D_CHIP	D_XOR	MES	SW
D_CPU	DAC_BRDG	MFB	TRA
D_CPUPIC	DAC_HIZ	MIF	TRIANGLE
D_DFF	DEV	MOS1	URC
D_DLATCH	DIO	MOS2	VCCS
D_DT	DIVIDE	MOS3	VCVS
D_FDIV	ENH	MULT	VERIZA
D_INV	EVT	N1	VHDL2A
D_JKFF	EW_CAP	NCO	VHDL2HIZ
D_NAND	EW_IND	NOISE	VSRC
D_NOR	EW_NLA2D	ONESHOT	XCAP
	EW_NOISE		ZENER

model_name — is a valid SPICE identifier which will be used on SPICE deck .model records to refer to this code model. It may or may not be the same as the C function name.

text — is a string describing the purpose and function of the code model.

For example:

```
NAME_TABLE:

Spice_Model_Name: capacitor
C_Function_Name: cm_capacitor
```

```
Description:   "Capacitor with voltage initial
condition"
```

Port Table

The device ports are defined in the port tables. The port table has the following syntax:

```
PORT_TABLE:

Port_Name:    name
Description:  text
Default_Type:    default
Allowed_Type:    [type type type]
Vector:          vector
Vector_Bounds:   size
Direction:       dataflow
Null_Allowed:    null
```

where:

name — is a valid SPICE identifier giving the name of the port.

text — is a string describing the purpose and function of the port.

default — specifies the type used for the port when no type is explicitly specified. Must be one of the items listed in "type".

type — lists the allowed types to which the port can be connected, with names separated by commas or spaces (for example, [d, g, h].

Type Name	Valid Directions	Description
d	in, out	digital
g	in, out	resistance (current input, voltage output)
	in, out	differential conductance (voltage input, current output)
		conductance (voltage input, current output)
gd	in, out	
h		

	hd	in, out	differential resistance (current input, voltage output)
	i	in, out	current
	id	in, out	differential current
	v	in, out	voltage
	vd	in, out	differential voltage

vector — specifies whether or not port is a vector and can be considered a bus. Choose from:
- yes - this port is a vector
- no - this port is not a vector

size — for port that are vectors only, specifies upper and lower bounds on vector size. Lower bound specifies minimum number of elements, upper bound specifies maximum number of elements. For unconstrained range, or ports that are not a vector, use a hyphen ("-")

dataflow — specifies the dataflow direction through the port. Choose from:
- in
- out
- inout

null — specifies whether or not it is an error to leave the port unconnected. Choose from:
- yes - this port may be left unconnected
- no - this port must be connected

For example:

```
PORT_TABLE:

Port_Name:        cap
Description:      "capacitor terminals"
Direction:        inout
Default_Type:     hd
Allowed_Types:    [hd]
Vector:           no
Vector_Bounds:    -
Null_Allowed:     no
```

Parameter Table

The device parameters are defined in the parameter tables. The parameter table has the following syntax:

```
PARAMETER_TABLE:

Parameter_Name:  name
Description:     text
Data_Type:       type
Vector:          vector
Vector_Bounds:   size
Default_Value:   default
Limits:          range
Null_Allowed:    null
```

where:

name	is a valid SPICE identifier which will be used on SPICE deck .model cards to refer to this parameter
text	is a string describing the purpose and function of the parameter
type	is the parameter data type. Corresponds to the underlying C data type (e.g. “double”), not the conceptual type of the parameter (e.g. “voltage”). Choose from: - boolean (if C data type is “Boolean_t” with valid values MIF_TRUE and MIF_FALSE) - complex (if C data type is “Complex_t” with double members real and imag) - int (if C data type is “int”) - real (if C data type is “double”) - string (if C data type is “char*”) - pointer (if C data type is “void*”)
vector	specifies whether parameter is vector or scalar. Choose from: - yes - parameter is vector - no - parameter is scalar
size	for parameters that are vectors only, specifies upper and lower bounds on vector size. Lower bound specifies minimum number of elements, upper bound specifies maximum number of elements. For unconstrained range, or parameters that are not a vector, use a hyphen (“-”). Alternatively, specifies the name of the port whose vector size is to be used for this parameter.

default	if Null_Allowed is “yes”, a default value to be used if the SPICE deck .model line does not supply a value for the parameter. Value must correspond to Data_Type (numeric, boolean, complex or string literal).
range	is a limited range of values (for “int” and “real” type parameters only).
null	specify whether or not parameter is allowed to be null. Choose from: - yes - yes - the corresponding SPICE deck .model card may omit a value for this parameter, and the default value will be used or, if no default value, an undefined value will be passed to the code model - no - no - this parameter must have a value. XSPICE will flag an error if the corresponding SPICE deck .model card omits a value for this parameter.

For example:

```
PARAMETER_TABLE:

Parameter_Name:        c                ic
Description:          “capacitance”    “voltage initial
condition”
Data_Type:             real             real
Default_Value:         -                0.0
Limits:                -                -
Vector:                 no               no
Vector_Bounds:          -                -
Null_Allowed:          no               no
```

Example Interface File

Here is an example interface file:

```
/* =======================================================
FILE    ifspec.ifs
MEMBER OF process XSPICE
Copyright 1991
Georgia Tech Research Corporation
Atlanta, Georgia 30332
All Rights Reserved

PROJECT A-8503

AUTHORS
    9/12/91  Bill Kuhn

MODIFICATIONS
    <date> <person name> <nature of modifications>
```

```
SUMMARY
This file contains the definition of a capacitor code
model with voltage type initial conditions.

INTERFACES
None.

REFERENCED FILES
None.

NON-STANDARD FEATURES
None.
================================================= */

NAME_TABLE:

Spice_Model_Name:      capacitor
C_Function_Name:       cm_capacitor
Description:           "Capacitor with voltage initial
condition"


PORT_TABLE:

Port_Name:             cap
Description:           "capacitor terminals"
Direction:             inout
Default_Type:          hd
Allowed_Types:         [hd]
Vector:                 no
Vector_Bounds:          -
Null_Allowed:          no


PARAMETER_TABLE:

Parameter_Name:        c                ic
Description:          "capacitance"    "voltage initial
condition"
Data_Type:             real             real
Default_Value:         -                0.0
Limits:                -                -
Vector:                 no               no
Vector_Bounds:          -                -
Null_Allowed:          no               no
```

The Implementation File (Cfunc.mod)

At each simulation iteration for a circuit using the code model, Multisim's XSpice simulation engine calls the implementation file. An example of an implementation file is shown in the *Example Implementation File* section. The implementation file, along with the interface file, needs to be coupled into a DLL to complete the code model.

The code model function then generates the code-modeled device's output. This output is based on the following:

- The input that XSpice presents to the code model function.
- The state of the model, which is stored and returned by XSpice.

The implementation file includes one or more of the macros, shown in the *Implementation File C Macros* section, that provide the API (Application Programming Interface) between XSpice and the code model.

This section lists the macros from which you can select. The example file shown in the *Example Implementation File* section gives an example of how to implement a macro. The implementation file, along with the interface file, needs to be compiled into a DLL to complete the code model.

Implementation File C Macros

AC_GAIN(outputname, inputname)

Type	Complex_t
Args	y[i], x[i]
Applies to	Analog code models only (event-driven or digital code models should do nothing during AC analysis).
Description	Assigns a value to this macro to specify the gain from outputname to inputname at the current frequency. The code model function is called once for each frequency point simulated.

ANALYSIS

Type	enum
Args	none

Applies to	All code models, since their behavior typically changes depending on the type of analysis being performed, and this macro can be used to specify appropriate output macros.
Description	Returns the type of analysis being performed: MIF_AC for AC MIF_DC for DC operating point MID_TRAN for transient

ARGS

Type	Mif_Private_t
Args	none
Applies to	All code models.
Description	The code model function's parameter list. Must be present and should not be modified.

CALL_TYPE

Type	enum
Args	none
Applies to	Only code models that are mixed-mode (analog and event-driven or digital).
Description	If the analog portion of the simulator requested the code model call, set to MIF_ANALOG. If the digital portion of the simulator requested the code model call, set to MIF_EVENT. Needed if a code model's computation effort can be reduced based on the type of call made.

INIT

Type	Boolean_t
Args	none
Applies to	All code models.
Description	If this is the first call to the code model function during the current analysis or batch of analyses, set to MIF_TRUE. Otherwise, set to MIF_FALSE. Needed to let the code model perform startup activities (for example, allocated memory) at the start of simulation only.

INPUT(inputname)

Type	double or void *
Args	name [i]
Applies to	Analog/mixed-mode code models.
Description	Only analog inputs are allowed (for event-driven, use INPUT_STATE and INPUT_STRENGTH). Returns the value on the node or branch connected to inputname. Type/units of input value is specified when input type is specified in the Ifspec.Ifs file.

INPUT_STATE(inputname)

Type	enum
Args	name [i]
Applies to	Digital/mixed-mode code models.
Description	Only event-driven/digital inputs are allowed (for analog, use INPUT). Returns the digital value (ZERO, ONE or UNKNOWN) at node at inputname. When a single output is connected to that node, this will equal the value of the last output event. When multiple outputs are connected, conflict resolution is performed.

INPUT_STRENGTH(inputname)

Type	enum
Args	name [i]
Applies to	Digital/mixed-mode code models.
Description	Only event-driven/digital inputs are allowed (for analog, use INPUT). Returns the digital strength (STRONG, RESISTIVE, HI_IMPEDANCE or UNDETERMINED) of node at inputname. When a single output is connected to that node, this will equal the strength of the last output event. When multiple outputs are connected, conflict resolution is performed.

INPUT_TYPE(inputname)

Type	char *
Args	name [i]

Applies to All code models.

Description Any inputs allowed. Returns the type string (i.e.: “v” for voltage, “i” for digital, “hd” for differential conductance, etc.) which describes the current usage of inputname. Needed to distinguish between “simulation time” usage of an input or output with more than one allowed type. For example, used for an input which has allowed types [v, i] and behaves differently when the input is voltage vs. current.

LOAD(inputname)

Type double

Args name [i]

Applies to Digital/mixed-mode code models.

Description Only event-driven/digital inputs are allowed. Assign a value to LOAD to set the input load due to inputname on the connected node. The load is given as a capacitance (normalized to 1ohm resistance) which is summed with all the other loads on the event-driven node to yield the total delay of the node.

MESSAGE(outputname)

Type char *

Args name [i]

Applies to Digital/mixed-mode code models.

Description Only event-driven/digital outputs are allowed. A message string to be placed on an event-driven node can be assigned to MESSAGE. Allows a code model to issue a message associated with a node.

OUTPUT(outputname)

Type double or void *

Args name [i]

Applies to Analog/mixed-mode code models.

Description Only analog outputs are allowed (for event-driven, use OUTPUT_STATE and OUTPUT_STRENGTH and OUTPUT_DELAY). Assigns a value to the node or branch connected to outputname. Type/units of output value specified when output type is specified in the Ifspec.Ifs file.

OUTPUT_CHANGED(outputname)

Type Boolean_t
Args name [i]
Applies to Digital/mixed-mode code models.
Description Only event-driven/digital inputs are allowed. Set to MIF_TRUE by default. Assign MIF_FALSE to indicate no change on that output. Allows the code model to specify that the event-driven output did not change and thereby speed up simulation.

OUTPUT_DELAY(outputname)

Type none
Args double
Applies to Digital/mixed-mode code models.
Description Only event-driven/digital inputs are allowed (for analog, use OUTPUT).
Sets the delay after which the transition event specified by OUTPUT_STATE occurs.

OUTPUT_STATE(outputname)

Type none
Args Digital_State_t
Applies to Digital/mixed-mode code models.
Description Only event-driven/digital outputs are allowed (for analog, use OUTPUT). Assigns the digital value (ZERO, ONE or UNKNOWN) to node at outputname by creating an event which is a transition to that value. When a single output is connected to that node, this will equal the value of the last output event. When multiple outputs are connected, conflict resolution is performed.

OUTPUT_STRENGTH(outputname)

Type	none
Args	Digital_State_t
Applies to	Digital/mixed-mode code models.
Description	Only event-driven/digital outputs are allowed (for analog, use OUTPUT). Assigns the digital strength (STRONG, RESISTIVE, HI_IMPEDANCE or UNDETERMINED) at node at outputname. When a single output is connected to that node, this will equal the strength of the last output event. When multiple outputs are connected, conflict resolution is performed.

OUTPUT_TYPE(inputname)

Type	char *
Args	name [i]
Applies to	Digital/mixed-mode code models.
Description	Any output allowed. Returns the type string (i.e.: “v” for voltage, “i” for digital, “hd” for differential conductance, etc.) which describes the current usage of outputname. Needed to distinguish between “simulation time” usage of an input or output with more than one allowed type. For example, used for an input which has allowed types [v, i] and behaves differently when the input is voltage vs. current.

PARAM(paramname)

Type	CD
Args	name [i]
Applies to	Any code model.
Description	Applies to all parameters. Returns the value paramname. Needed to access model parameters specified in the netlist.

PARAM_NULL(paramname)

Type	Boolean_t
Args	name [i]

Applies to	Only parameters allowed to be unspecified (Null allowed in the param table of the Ifspec.Ifs file is yes).
Description	Returns MIF_TRUE if paramname was not specified in the netlist and MIF_FALSE if it was specified. Allows the code model to tell if a parameter value equals its default because the default value was actually specified.

PARAM_SIZE(paramname)

Type	int
Args	name
Applies to	Vector type parameters only.
Description	Returns the number of elements in a vector type parameter. Needed to iterate over the vector parameter if the number of vector elements is not fixed.

PARTIAL

Type	double
Args	y[i], x[i]
Applies to	Analog/mixed-mode code models.
Description	Partial derivative of output y with respect to input x.

PORT_NULL

Type	Boolean_t
Args	name[i]
Applies to	Any code model.
Description	Has this port been specified as unconnected?

PORT_SIZE

Type	int
Args	name
Applies to	Any code model.
Description	Size of port vector.

RAD_FREQ

Type	double
Args	<none>
Applies to	Analog/mixed-mode code models.
Description	Current analysis frequency in radians per second.

T (<n>)

Type	double
Args	<none>
Applies to	All code models.
Description	History of the previous nth analysis time (TIME = T[0]). Maximum of 8.

TEMPERATURE

Type	double
Args	<none>
Applies to	All code models.
Description	Current analysis temperature.

TIME

Type	double
Args	<none>
Applies to	All code models.
Description	Current analysis time (same as T[0]).

Example Implementation File

Here is an example implementation file:

```
/* ================================================
FILE    cfunc.mod

MEMBER OF process XSPICE

Copyright 1991
Georgia Tech Research Corporation
Atlanta, Georgia 30332
All Rights Reserved
```

```
PROJECT A-8503

AUTHORS
    9/12/91  Bill Kuhn

MODIFICATIONS
    <date> <person name> <nature of modifications>

SUMMARY
   This file contains the definition of a capacitor code
model
    with voltage type initial conditions.

INTERFACES
    cm_capacitor()

REFERENCED FILES
    None.

NON-STANDARD FEATURES
    None.

========================================================= */

#define VC  0


void cm_capacitor (ARGS)
{
    Complex_t   ac_gain;
    double      partial;
    double      ramp_factor;
    double      *vc;


    /* Get the ramp factor from the .option ramptime */
    ramp_factor = cm_analog_ramp_factor(MIF_INSTANCE);

    /* Initialize/access instance specific storage for
capacitor volt age */
    if(INIT) {
      cm_analog_alloc(MIF_INSTANCE,VC, sizeof(double));
        vc = cm_analog_get_ptr(MIF_INSTANCE,VC, 0);
        *vc = PARAM(ic) *
cm_analog_ramp_factor(MIF_INSTANCE);
    }
    else {
```

```
            vc = cm_analog_get_ptr(MIF_INSTANCE,VC, 0);
        }

        /* Compute the output */
        if(ANALYSIS == DC) {
            OUTPUT(cap) = PARAM(ic) * ramp_factor;
            PARTIAL(cap, cap) = 0.0;
        }
        else if(ANALYSIS == AC) {
            ac_gain.real = 0.0;
            ac_gain.imag = -1.0 / RAD_FREQ / PARAM(c);
            AC_GAIN(cap, cap) = ac_gain;
        }
        else if(ANALYSIS == TRANSIENT) {
            if(ramp_factor < 1.0) {
                *vc = PARAM(ic) * ramp_factor;
                OUTPUT(cap) = *vc;
                PARTIAL(cap, cap) = 0.0;
            }
            else {
               cm_analog_integrate(MIF_INSTANCE,INPUT(cap)
/ PARAM(c),
                vc, &partial);
                partial /= PARAM(c);
                OUTPUT(cap) = *vc;
                PARTIAL(cap, cap) = partial;
            }
        }
}
```

7

Multisim SPICE Reference

The following sections document SPICE-based circuit syntax that is supported by Multisim's simulation engine. The sections describe general purpose syntax used for such operations as device declaration, and device-specific syntax used to parameterize primitive devices such as MOSFETs.

The scope of this section is to serve as a reference guide. For more information about SPICE, you may wish to consult The SPICE Book, Andrei Vladimirescu, John Wiley & Sons Inc., 1994, ISBN-13: 978-0-471-60926-1, or Semiconductor Device Modeling with SPICE, Second Edition, Giuseppe Massobrio and Paolo Antognetti, McGraw-Hill, 1993, ISBN-13: 978-0070024694.

Note Since the simulation engine is case insensitive, the use of both upper and lower case characters in this document is done strictly for clarity.

Documentation Conventions

This specification uses the following conventions:

- Text enclosed in <> is optional.
- Red text is a variable, a function, or an instance name.
- Orange text is a mathematical expression or numeric constant.
- Blue text is a node identifier.
- Green text is a model identifier, device name, or subckt name.
- *Blue italic text* is an example.
- Brown text is an XSPICE terminal type specifier.

General Purpose Syntax

This section describes the overall syntax and building blocks of SPICE models.

Primitive Device Declarations
SPICE Subcircuits
Circuit Parameters
Number Format
Comments and Line Continuation

Primitive Device Declarations

A primitive device is the lowest level model that can be used in a circuit and is a building block for macromodels and entire circuits. Multisim supports many such devices. Refer to the *Analog Devices* section for more information.

This section looks at how a primitive device is declared and used in a circuit.

Primitive devices are comprised of either just an instance declaration or an instance declaration with an associated model definition.

Instance Declaration

The instance declaration places a primitive device between circuit nodes, specifies device parameters, and links the instance to a model definition (where needed). The instance declaration has the following general format:

```
PREFIX_anyname node1 <node2 <…>> my_ModelNAME
Instance_line_parameters
```

Identifier	Description
PREFIX	Device specific character.
_anyname	Arbitrary instance name suffix.
nodeN *where N=1,2,3…*	The name of the Nth node that the device is connected to. Node names may contain any characters except for white space and the following: \ {} () [] : # " ' ; , % < > ` & = *

Identifier	Description
my_ModelNAME	Arbitrary name that links the instance declaration to a model definition. It is optional for some devices is and mandatory for others.
Instance_line_parameters	A list of device specific instance parameters, some of which are mandatory and some optional.

Model Definition

Required for some primitive devices and optional for others, the model line allows users to specify additional parameters for a device. The model line has the following format:

```
.model my_ModelNAME <AKO:akomodelname>
devicename(<Model_Parameters>)
```

Identifier	Description
my_ModelNAME	Arbitrary model identifier that links the model definition to one or more instance declarations.
akomodelname	Model identifier from which specified model parameters will be inherited.
devicename	Device-specific identifier.
Model_Parameters	List of device-specific model parameter assignments. Note that every model parameter of every device has a default value. Assignments made in the *Model_Parameters* block overwrite these defaults.

All device-specific information regarding prefix characters, number of circuit nodes, model definition requirements, and device parameters is contained in the *Analog Devices* section.

Examples

```
*Resistor device without its optional model line
R88 1 0 10k tc1=0.1 tc2=5

*Resistor device with its optional model line
R15 in8 out myResmodel 10k
.model myResmodel res(rmult=5)

*A BJT with area factor of 8
*and BJT with an area factor 4
Q54 99 b 0 BJTer area=8
Q55 e b 0 BJTer area=4
.model BJTer npn(is=1e-12 bf=140 rc=3 tf=3n)

*MOSFET M1 will inherit
*kp=1e-3 phi=0.76 mj=0.44 ld=0.1u
M1 1 4 9 9 myMOS1
.model myMOS1 AKO:myMOS2 nmos(vto=1.4)
.model myMOS2 nmos(kp=1e-3 phi=0.76 mj=0.44 ld=0.1u)
```

SPICE Subcircuits

A SPICE subcircuit wraps around a block of circuit text and allows external connections to this circuitry only through the subcircuit's port. The benefit of this is that the internal circuitry is isolated from external circuitry, thus internal devices and node names with the same names as those external to the subcircuit are neither conflicting nor shorted. In addition, subcircuits can accept circuit parameters which can be used to assign values to internal devices or nested subcircuits.

A subcircuit is an extremely useful concept, forming the basis of any modular or hierarchical design.

Subcircuit Definition

```
.SUBCKT mysubcktname node1 <node2 <…>> <OPTIONAL:
optionalnode1=defaultnode1 <optionalnode2=defaultnode2
<…>>> <PARAMS: param1=default1 <param2=default2 <…>>>
<Subcircuit Contents>
.ENDS
```

Identifier	Description
mysubcktname	Arbitrary subcircuit identifier that links the subcircuit to an instance declaration.
node*N* *where N=1,2,3…*	The name of the Nth node for the subcircuit port. These are the nodes that Multisim's component symbol pins would be mapped to. Refer to the *Advanced Pin Mapping Dialog* section of Chapter 6, *Component Editing* for more information.
optionalnode*N* *where N=1,2,3…*	The name of the Nth optional node for the subcircuit port. These nodes are optionally included on the instance line and if they are not provided then they will be connected to optionalnode*N*. Optional nodes are not available on Multisim components and thus they are not useful on the top level models of a component.
defaultnode*N* *where N=1,2,3…*	The default node to connect to for optionalnodeN if that node is not provided on the instance line. This node should be a valid node from outside this subcircuit declaration.
param*N* *where N=1,2,3…*	The name of the Nth input circuit parameter for the subcircuit.
param*N* *where N=1,2,3…*	A default value for the Nth input parameter if PARAM*N* is not specified on the subcircuit instance declaration.

A subcircuit definition is only useful if it is referenced by one or more instance declarations inside a circuit. This can be done as a top level model of a component on a schematic or by declaring an instance of the subcircuit within a model.

Instance Declaration

```
Xanyname node1 <node2 <…>> mysubcktName <PARAMS:
PARAM1=expression1 <PARAM2=expression2 <…>>>
```

Identifier	Description
nodeN *where N=1,2,3…*	The Nth node for the subckt. The number of nodes must match the number of nodes in the subcircuit definition.
mysubcktName	An arbitrary subcircuit identifier linking the instance declaration to a subcircuit.
param*N* *where N=1,2,3…*	The name of the input parameter for the subckt. The nodes do not have to be in the same order as the subcircuit definition and do not all need to be present. If any particular PARAM*N* is omitted then the subcircuit will use default*N*.
expression*N* *where N=1,2,3…*	The expression for the Nth input parameter. See the expression section for details regarding expressions.

Additional Notes

- This SPICE-based subcircuit should not be confused with a Multisim schematic capture Subcircuit which is used to created hierarchy with schematic symbols.
- If the “PARAMS:” keyword is omitted within the circuit parameters portion of the declaration, the entire circuit parameters portion must be enclosed by round “( )” or curly “{ }” parenthesis.
- Node “0” is a global node - regardless of circuit or subcircuit hierarchy, all nodes with the name “0” are connected together.

Examples

```
*A resistive voltage divider circuit that uses a resistor
subcircuit model. The upper resistor is 10k and the lower
is 47k
V1 in 0 10
X1 in mid res_block params: res_val=10k
X2 mid 0 res_block
.subckt res_block 1 2 params: res_val=47k
```

```
R1 1 2 {res_val}
.ends

*expression usage
.param gain=100
V1 5 0 3.3
X1 5 8 AMP PARAMS: ampgain={limit(gain, 200, 80)}
.subckt AMP in out PARAMS: ampgain=90
E1 out 0 in 0 {ampgain}
.ends
```

Circuit Parameters

Circuit parameters allow flexibility in assigning device parameters. The general format for defining a circuit parameter is as follows:

```
.PARAM my_parameter_name = expression
```

Identifier	Description
my_parameter_name	Arbitrary parameter name. It may contain numbers, letters, and underscores but not any other symbols. In addition, it must not start with a number.
expression	Arbitrary expression operating on numerical constants or other circuit parameters. Refer to the *Mathematical Expressions* section for more details regarding expressions. Circuit variables (node voltage and device current) are not permitted.

Additional Notes

- Circuit parameters are constants, and thus may not contain any circuit variables such as node voltages or branch currents.
- Circuit parameters take precedence over predefined constants. If you define a variable called pi it will be used in place of the normal built-in constant.
- Parameter variables exist at the level of the circuit they are defined at, so if a parameter is defined in the main circuit it can also be used in the SPICE subckts. However, if it is defined in a SPICE subckt, then that parameter will be meaningless in the circuit levels above. Expressions always use the most locally defined value. In the example below, the resistor R1's value is 3.14, R2's value is 2, R3's value is 3, and R4's value is 4.

```
R1 51 0 {pi}
X1 1 2 mysub

.SUBCKT mysub node1 node2
      .PARAM pi = 2
      R2 node1 node2 { pi }
      X2 node3 node2 mysub2
      .SUBCKT mysub2 node1 node2
            .PARAM pi = 3
            R3 node1 node2 { pi }
            R4 node1 node2 {varb}
      .ENDS
      .PARAM varb = 4
.ENDS
```

Examples

```
.PARAM a = 6
.PARAM n = {0.5}
.PARAM maxV = a+10

V1 in 0 {maxV}
D1 in 3 mydiode
.model mydiode d(n={n+0.01})
```

Number Format

Numbers are standard floating point or integer numbers with optional suffixes. Any characters that follow directly, unless they are mathematical operators, are ignored.

Number suffixes:

Suffix	Meaning	Multiplier
f, F	fento	1e-15
p, P	pico	1e-12
n, N	nano	1e-9
u, U	micro	1e-6
m, M	milli	1e-3
k, K	Kilo	1e3
meg, MEG	Mega	1e6

Suffix	Meaning	Multiplier
g, G	Giga	1e9
t, T	Tera	1e12

Examples

```
R1 1 2 10mohm

*the following voltage source has a DC value of
0.00000521
V1 4 0 {0.21e-6+5uV}
```

Comments and Line Continuation

The asterisk (*) and the semicolon (;) characters can be used to comment out individual line of circuit text.

The semicolon (;) is used within a line of text to comment out everything to the right.

A plus (+) is used to continue a SPICE statement from the previous line.

The comment works on a per line basis—a single asterisk does not comment out an entire SPICE statement spread over multiple lines.

Examples

```
*This example shows how comments work
V1 1 0 10 ;V1 is in the circuit, but this is just a
comment
R1 1 2 10k
*Below, Resistor R2 is taken out of the circuit
*R2 1 2 10k

*In the following, we only eliminate the 'n' parameter
from the multiline SPICE statement
D1 2 0 myDiode
.model myDiode d(Is=1e-12
*+n=1.2
+rs=3.7)
```

Mathematical Expressions

Introduction
Supported Mathematical Functions, Operators and Constants
User-Defined Functions

Introduction

Mathematical expressions allow the user to create arbitrary mathematical expressions consisting of various functions and operators and apply the results to device parameters. Expressions are very useful modeling tools when used within the Arbitrary Source devices. Refer to the *Arbitrary Source* section for more information.

The functions and operators in the expression can operate on numerical constants, on circuit parameters, and, when used within the Arbitrary Source device, on live circuit variables. Refer to the *Supported Mathematical Functions, Operators and Constants* section for a list of supported mathematical functions, operators and pre-defined constants.

Within the Arbitrary Source device, special functions `V(nodeabs)`, `V(node+,node-)`, and `I(deviceX)` can be used to reference the circuit voltages and currents.

- `V(nodeabs)` references the voltage at node nodeabs relative to ground.
- `V(node+,node-)` references the difference between node+ and node-.
- `I(deviceX)` references the current through the device with the instance name `deviceX`. Currently only the Independent Voltage source, dependent voltage sources, and inductor devices are supported.

Additional Notes

- We recommend that expressions be enclosed in { } to avoid ambiguous syntax. However, this is not required for simulation.
- Portions of expressions that are not enclosed in { } or () should not contain any spaces.

Examples

```
*High-level filter specification
.param pole=1k
.param res_val=1k
R1 in out {res_val}
C1 out 0 {1/(2*pi*res_val*pole)}

*A very simple diode modeled using an expression in an
Arbitrary source
G1 A C value={1e-12*(e^(V(A,C)/0.025)-1 ) }
```

Supported Mathematical Functions, Operators and Constants

Mathematical functions:

Function Name	Alternate Name	Parameters	Description	Notes
if	—	(test,a,b)	If-else function. If the test returns true, the result is a, else it returns b.	Example: B1 out 0 V={if(v(1)>5, v(1)**2, 0)}
sin	sine	(x)	Sine function.	—
asin	arcsin	(x)	Arc-sine function.	—
sinh	—	(x)	Hyperbolic sine function.	—
asinh	arcsinh	(x)	Arc-hyperbolic sine function.	—
cos	—	(x)	Cosine function.	—
acos	arccos	(x)	Arc-cosine function.	—
cosh	—	(x)	Hyperbolic cosine function.	—
acosh	arccosh	(x)	Arc-hyperbolic cosine function.	—
tan	—	(x)	Tangent function.	—
tanh	—	(x)	Hyperbolic tangent function.	—

Function Name	Alternate Name	Parameters	Description	Notes
atan	arctan	(x)	Arc-tangent function.	—
atanh	arctanh	(x)	Arc-hyperbolic tangent function.	—
atan2	—	(x, y)	Atan 2 function.	Same as: atan (x/y)
exp	—	(x)	Calculates the exponential e^x.	—
exp1	—	(x, y)	Calculates the exponential with a maximum value.	Same as: min (exp (x), y)
ln	—	(x)	The natural logarithm function.	—
log10		(x)	The base-10 logarithm function.	
log		(x)	The generic logarithm function. This function is included for legacy compatibility and its use is not recommended.	Normally this is the same as ln(x) except in 'B' arbitrary sources where it is log10(x). This behavior is also changed by the .SYNTAX statements.
sqrt	—	(x)	Square root function.	—
abs	—	(x)	Absolute value function.	—

sgn	sign	(x)	Sign or signum function.	if (x < 0)
				sgn(x) ==–1
				if (x > 0)
				sgn(x) == 1
				if (x == 0)
				sgn(x) == 0

max	—	(x, y)	Returns the maximum of x and y.	—

Function Name	Alternate Name	Parameters	Description	Notes
min	—	(x, y)	Returns the minimum of x and y.	—

uramp	—	(x)	Ramp function, clips the value against a minimum of 0.	if (x < 0)
				uramp(x) == 0
				if (x >= 0)
				uramp = x

u	stp, step	(x)	Step function.	if (x < 0)
				u(x) == 0
				if (x > 0)
				u(x) == 1
				if (x == 0)
				u(x) == 0.5

table	—	(expr, x1,y1, <x2,y2 <…>>)	Table function. This is the function version of the TABLE based Arbitrary source device.	Refer to the *Arbitrary Source* section for more information.
limit	—	(x, a, b)	Clips the input value x to the range (A,B).	—
pwr	—	(x, y)	The pwr function.	Same as: abs(x)^y

pwrs	—	(x, y)	The pwrs function.	if (x < 0)
				pwrs(x) == –(x**y)
				if (x >= 0)
				pwrs(x) == (x**y)

V	—	(node)	Voltage of a node.	Can only be used in non-linear controlled source expressions.

Function Name	Alternate Name	Parameters	Description	Notes
V	—	(node1, node2)	Voltage difference of two nodes.	Can only be used in non-linear controlled source expressions. Same as: V(node1)-V(node2) Example: E1 out 0 value={V(1,2)*15}
I	—	(voltage source) or (inductor)	Current through a voltage source or an inductor.	Can only be used in non-linear controlled source expressions. Example: E1 out 0 value={I(Vsense)**2 + I(E99)}

positive	—	(x)	Ensure positive function.	if (x < d)
				positive(x) = d
				else
				positive(x) = x
				where d is 1.0p

negative	—	(x)	Ensure negative function.	if (x >–d)
				negative(x) =–d
				else
				negative(x) = x
				where d is 1.0p

nonpos	—	(x)	Ensure not-positive function.	if (x > 0)
				nonpos(x) = 0
				else
				nonpos(x) = x

nonneg	—	(x)	Ensure not-negative function.	if (x < 0)
				nonneg(x) = 0
				else
				nonneg(x) = x

Function Name	Alternate Name	Parameters	Description	Notes

nonzero	—	(x)	Ensure non-zero function.	if (x < d) & (x >= 0)
				nonzero(x) = d
				if (x >–d) & (x <= 0)
				nonzero(x) =–d
				else
				nonzero(x) = x
				where d is 1.0p

zero	—	(x)	Evaluates x but always returns a value of 0.0.	—
one	—	(x)	Evaluates x but always returns a value of 1.0.	—
schedule	—	(expr, x1,y1, <x2,y2 <…>>)	Schedule function. Gives a value of yN for when time between xN and xN+1.	Time is simulation time.

Mathematical operators:

Symbol	Alternate Symbol	Description	Usage
+		Addition.	A + B
-		Subtraction.	A – B
/		Division.	A / B
*		Multiplication.	A * B
**	^	Exponentiation (power).	A ** B or A ^ B This can be interpreted differently in other simulators. Refer to the *Compatibility Modes* section for more information.
<		Less than.	A < B
<=		Less than or equal to.	A <= B
>		Greater than.	A > B

Symbol	Alternate Symbol	Description	Usage
>=		Greater than or equal to.	A >= B
==		Equal to.	A == B
!=		Not equal to.	A != B
&		Logical AND.	A & B
|		Logical OR.	A | B
xor		Logical XOR.	A XOR B
? :		Ternary if. This operator has two symbols and three operands.	A ? B : C This is the same as IF(A,B,C).

Built-in constants:

Symbol	Description	Value
true	Boolean true value.	1.0
false	Boolean false value.	0.0
yes	Alternate form of boolean true value.	1.0
no	Alternate form of boolean false value.	0.0
pi	The constant pi.	3.14159265358979323846
e	The constant e.	2.71828182844590452353
c	Speed of light.	2.99792458e8
kelvin	Constant to convert between degrees Kelvin and Celsius, and vice versa.	–273.0
echarge	Electron charge.	1.602176487e-19
boltz	Boltzmann’s constant.	1.3806503e-23
planck	Planck’s constant.	6.62606896e-34

Symbol	Description	Value
temp	Current temperature of the simulation in degrees Celsius.	The default is 27, but it can be changed from the simulation options.
time	Current time of the simulation in seconds.	Current time of the simulation in seconds (it is constant with respect to circuit variables).

Additional Notes

- IMPORTANT! Unlike most mathematical languages, Multisim considers the exponentiation (power) operators ** and ^ to be of higher precedence than unary minus. This means that {–5**2} is +25 while {0–5**2} is –25. Although this is unintuitive, it is standard among SPICE simulators. Use brackets to ensure logical, readable expressions in this case.

User-Defined Functions

Similar to the way subcircuits provide modularity to connecting together blocks of circuits, user-defined functions provide modularity for using mathematical expressions.

Declaration

```
.FUNC my_function_name( Arg1 <,Arg2 <…> > ) =
valueexpression
```

Identifier	Description
my_function_name	Arbitrary function name.
ArgN where N=1,2,3	List of arguments used by the function.
valueexpression	Mathematical Expression operating on the arguments

Additional Notes

- User-defined functions can be called from any field where a mathematical expression is used. However, only the Arbitrary Source device can call a function with circuit variables (node voltages and device currents) in the arguments.

- Function names may contain numbers, letters and underscores but must not start with a number and may not contain any other symbols.
- The value expression may be enclosed in { } for clarity, but this is not mandatory.
- User-defined functions take precedence over predefined functions; if you define a function called `sin` it will take the place of the standard sin function within that context.

Examples

```
.FUNC sinplusn ( angle, n ) = {sin(angle) + n}
.FUNC myfunc(a,b,c)=a+sinplusn(b**c,0.1)
.param foo=myfunc(1,2,3)

E1 60 59 value={5+myfunc(v(2),3*v(3),9)}
G1 0 88 value={myfunc(I(Vref), 1, 2}
```

Analog Devices

Resistor
Capacitor
Inductor
Coupled (Mutual) Inductor
Diode
Lossless Transmission Line
Lossy Transmission Line
Uniform R.C. Line (Lumped-approximation R.C. line)
JFET
MESFET
Voltage-Controlled Switch
Current-Controlled Switch
BJT
MOSFET
Independent Voltage Source
Independent Current Source
Arbitrary Source
Voltage Controlled Voltage Source
Current Controlled Voltage Source
Voltage Controlled Current Source
Current Controlled Current Source

Resistor

Resistor instance declaration syntax:

```
Rxxxx node1 node2 resistance <TC=tc1 <,tc2> >
<TEMP=temp>

Rxxxx node1 node2 resistance <TC1=tc1> <TC2=tc2>
<TEMP=temp>

Rxxxx node1 node2 <resistance> <Model> <L=l> <W=w>
```

Resistor instance declaration parameters:

Parameter Name	Parameter Description
resistance	Device resistance.
L	Device length.
W	Device width.
TC	Instance temperature coefficients. This is a two element vector for specifying TC1 and TC2.
TC1	Alternate way of specifying 1st-order temperature coefficient.
TC2	Alternate way of specifying 2nd-order temperature coefficient.
SENS_RESIST	[FLAG] Flag to request sensitivity with respect to resistance.
TEMP	Instance operating temperature.

The following only applies if a model has been specified in the instance declaration as it is not mandatory for resistors.

Resistor model definition syntax:

```
.MODEL mymodelname R ( <TC1=tc1 <TC2=tc2>>
<Other_Model_Parameters…> )

.MODEL mymodelname RES ( <TC1=tc1 < TC2=tc2>>
<Other_Model_Parameters…> )
```

Parameter Name	Parameter Description	Units	Default
DEFW	Default device width.	m	1×10^{-6}
NARROW	Narrowing due to side etching.	m	0.0
R	Resistance multiplier (same as RMULT).	—	1.0
RMULT	Resistance multiplier.	—	1.0
RSH	Sheet resistance.	Ω/sq.	0.0
TC1	Instance 1st-order temperature coefficient.	1/ºC	0.0
TC2	Instance 2nd-order temperature coefficient.	1/(ºC^2)	0.0
TCE	Exponential temperature coefficient.	%/(ºC^2)	0.0
TNOM	Temperature at which model parameters were measured.	ºC	—
T_ABS	Absolute temperature.	ºC	—
T_MEASURED	Temperature at which model parameters were measured.	ºC	—
T_REL_GLOBAL	Temperature delta relative to global temperature.	ºC	—
T_REL_LOCAL	Temperature delta relative to AKO model temperature.	ºC	—

Examples

```
R1 1 0 4.7k
R2 1 0 10k myRes
.model myRes r(tc1=1e-4)
```

Capacitor

Capacitor instance declaration syntax:

```
Cxxxx node1 node2 capacitance <IC=vc0> <TC=tc1 <,tc2> >

Cxxxx node1 node2 capacitance <Model> <L=L> <W=W>
<IC=vc0>
```

Capacitance instance declaration parameters:

Parameter Name	Parameter Description
capacitance	Device capacitance.
IC	Initial capacitor voltage.
TC	Instance temperature coefficients. This is a two element vector for specifying TC1 and TC2.
L	Device length.
W	Device width.
SENS_CAP	[FLAG] Flag to request sensitivity with respect to capacitance.
TEMP	Instance temperature.

The following only applies if a model has been specified in the instance declaration as it is not mandatory for capacitors.

Capacitor model definition syntax:

```
.MODEL mymodelname C ( <TC1=tc1 <TC2=tc2>> <VC1=vc1
<VC2=vc2>> + <Other_Model_Parameters…> )
```

Capacitor model definition parameters:

Parameter Name	Parameter Description	Units	Default
CJ	Junction bottom capacitance per area.	$F/(m^2)$	0.0
CJSW	Junction sidewall capacitance per meter.	F/m	0.0
CMULT	Capacitance multiplier.	—	1.0
DEFW	Default width.	m	$1x10^{-6}$
NARROW	Narrowing due to side etching.	m	0.0
TNOM	Temperature at which model parameters were measured.	ºC	—
T_ABS	Absolute temperature.	ºC	—

Parameter Name	Parameter Description	Units	Default
T_MEASURED	Temperature at which model parameters were measured.	ºC	—
T_REL_GLOBAL	Temperature delta relative to global temperature.	ºC	—
T_REL_LOCAL	Temperature delta relative to AKO model temperature.	ºC	—
TC1	1st-order temperature coefficient.	1/ºC	0.0
TC2	2nd-order temperature coefficient.	$1/(ºC^2)$	0.0
VC1	1st-order voltage coefficient.	1/V	0.0
VC2	2nd-order voltage coefficient.	$1/(V^2)$	0.0

Examples

```
c1 1 0 1u
C2 1 0 1e-12 myCap
.model myCap C(tc1=1e-4)
```

Inductor

Inductor instance declaration syntax:

```
Lxxxx node1 node2 <Model> inductance <IC=iL0>
```

Inductor instance declaration parameters:

Parameter Name	Parameter Description
inductance	Device inductance.
IC	Initial current through inductor.
SENS_IND	[FLAG] Flag to request sensitivity with respect to inductance.
TEMP	Instance temperature.

The following only applies if a model has been specified in the instance declaration as it is not mandatory for inductor.

Inductor model definition syntax:

```
.MODEL mymodelname L ( <Other_Model_Parameters…> )
.MODEL mymodelname IND ( <Other_Model_Parameters…> )
```

Inductor model definition parameters:

Parameter Name	Parameter Description	Units	Default
IL1	1st-order current coefficient.	1/A	0.0
IL2	2nd-order current coefficient.	$1/(A^2)$	0.0
LMULT	Inductance multiplier.	—	1.0
TNOM	Temperature at which model parameters were measured.	ºC	—
T_ABS	Absolute temperature.	ºC	—
T_MEASURED	Temperature at which model parameters were measured.	ºC	—
T_REL_GLOBAL	Temperature delta relative to global temperature.	ºC	—
T_REL_LOCAL	Temperature delta relative to AKO model temperature.	ºC	—
TC1	1st-order temperature coefficient.	1/ºC	0.0
TC2	2nd-order temperature coefficient.	$1/(ºC^2)$	0.0

Examples

```
L1 1 0 1m
l2 1 0 1e-12 myCap
.model myCap C(tc1=1e-4)
```

Coupled (Mutual) Inductor

Coupled inductor instance declaration syntax:

```
Kxxxx Lname1 Lname2 k
```

Coupled inductor instance declaration parameters:

Parameter Name	Parameter Description
k	Mutual inductance coefficient (0.0 to 1.0).
Lname1	Name of first coupled inductor.
Lname2	Name of second coupled inductor.

Examples

```
L1 1 0 1m
L2 77 0 1m
L3 88 0 1m
K1 L1 L2 0.99
K2 L1 L3 0.99
K3 L2 L3 0.99
```

Diode

Diode instance declaration syntax:

```
Dxxx nodeN+ nodeN- Model <area> <OFF> <IC=Vd0>
<TEMP=temp>
```

Diode instance declaration parameters:

Parameter Name	Parameter Description
area	Area factor.
OFF	Initially off.
IC	Initial device voltage.
TEMP	Instance temperature.

Diode model definition syntax:

```
.MODEL mymodelname D ( <Other_Model_Parameters…> )
```

Diode model definition parameters:

Parameter Name	Parameter Description	Units	Default
AF	Flicker noise exponent.	—	1.0
BV	Reverse breakdown voltage.	V	infinite
CJO	Junction capacitance.	F	0.0
EG	Bandgap voltage.	eV	1.11
FC	Forward bias junction fit parameter.	—	0.5
IBV	Current at reverse breakdown voltage.	A	1×10^{-10}
IBVL	Low-level reverse breakdown knee current.	A	0.0
IKF	High-injection knee current.	A	infinite
IS	Saturation current.	A	1×10^{-14}
ISR	Recombination currrent parameter.	A	0.0
KF	Flicker noise coefficient.	—	0.0
M	Grading coefficient.	—	0.5
N	Emission coefficient.	—	1.0
NBV	Reverse breakdown ideality factor.	—	1.0
NBVL	Low-level reverse breakdown ideality factor.	—	1.0
NR	Emission coefficient for ISR.	—	2.0
RS	Ohmic resistance.	Ω	0.0
TBV1	Linear BV temperature coefficient.	1/ºC	0.0
TBV2	Quadratic BV temperature coefficient.	1/(ºC^2)	0.0
TIKF	Linear IKF temperature coefficient.	1/ºC	0.0
TNOM	Parameter measurement temperature.	ºC	—
TRS1	Linear RS temperature coefficient.	1/ºC	0.0
TRS2	Quadratic RS temperature coefficient.	1/(ºC^2)	0.0
TT	Transit time.	s	0.0

Parameter Name	Parameter Description	Units	Default
VJ	Junction potential.	V	1.0
XTI	Saturation current temperature exponent.	—	3.0

Examples

```
*diode with area scale of 2
d1 1 0 myDiode 2
.model myDiode d(is=1.1p)
```

Lossless Transmission Line

Lossless transmission line instance declaration syntax:

```
Txxx nodeP1+ nodeP1- nodeP2+ nodeP2- Z0=z0 <TD=td>
<F=freq <NL=nl> <IC=v1 <,i1 <,v2 <,i2>>>> <REL=rel>
<ABS=abs>

Txxx nodeP1+ nodeP1- nodeP2+ nodeP2- Z0=z0 <TD=td>
<F=freq <NL=nl> <V1=v1> <I1=i1> <V2=v2> <I2=i2>
<REL=rel> <ABS=abs>
```

Lossless Transmission Line instance declaration parameters:

Parameter Name	Parameter Description
ZO	Characteristic impedance.
TD	Transmission delay.
F	Frequency.
NL	Normalized length a frequency given.
v1	Initial voltage v1.
i1	Initial current i1.
v2	Initial voltage v2.
i2	Initial current i2.
REL	Relative rate of change of derivative for breakpoint.
ABS	Absolute rate of change of derivative for breakpoint.

This device does not have an associated model definition.

Example

```
T1 1 0 2 0 Z0=75
```

Lossy Transmission Line

Lossy Transmission Line instance declaration syntax:

```
Oxxx node1 node2 node3 node4 Model <IC=v1 <,i1 <,v2
<,i2>>>>

Oxxx node1 node2 node3 node4 Model <V1=v1> <I1=i1>
<V2=v2> <I2=i2>
```

Lossy Transmission Line instance declaration parameters:

Parameter Name	Parameter Description
v1	Initial voltage at terminal 1.
v2	Initial voltage at terminal 2.
i1	Initial current at terminal 1.
i2	Initial current at terminal 2.

Lossy Transmission Line model definition syntax:

```
.MODEL mymodelname LTRA ( <NOCONTROL>
<STEPLIMIT/NOSTEPLIMIT>
+ <LININTERP/QUADINTERP/MIXEDINTERP>
<Other_Model_Parameters…> )
```

Lossy Transmission Line model definition parameters:

Parameter Name	Parameter Description	Units	Default
C	Capacitance per meter.	F/m	0.0
G	Conductance per meter.	S/m	0.0
L	Inductance per meter.	H/m	0.0
R	Resistance per meter.	Ω/m	0.0
LEN	Length of line.	—	required

Parameter Name	Parameter Description	Units	Default
REL	Relative rate of change of derivative for breakpoint.	—	1.0
ABS	Absolute rate of change of derivative for breakpoint.	—	1.0
NOCONTROL	[FLAG] No timestep control.	—	—
STEPLIMIT	[FLAG] Always limit timestep to 0.8*(delay of line).	—	—
NOSTEPLIMIT	[FLAG] Don't always limit timestep to 0.8*(delay of line).	—	—
LININTERP	[FLAG] Use linear interpolation.	—	—
QUADINTERP	[FLAG] Use quadratic interpolation.	—	—
MIXEDINTERP	[FLAG] Use linear interpolation if quadratic results look unacceptable.	—	—
TRUNCNR	Use N-R iterations for step calculation in LTRAtrunc.	—	—
TRUNCDONTCUT	Don't limit timestep to keep impulse response calculation errors low.	—	—
COMPACTREL	Special reltol for straight line checking.	—	—
COMPACTABS	Special abstol for straight line checking.	—	—

Example

```
O1 1 0 2 0 myLossyLine
.model myLossyLine LTRA(r=3.5 L=3m g=1e-6 c=3.2e-6)
```

Uniform R.C. Line (Lumped-approximation R.C. line)

Uniform R.C. line instance declaration syntax:

```
Uxxx node1 node2 nodeRef Model L=len <N=lumps>
```

Uniform R.C. line instance declaration parameters:

Parameter Name	Parameter Description
L	Length of transmission line.
N	Number of lumps.

Uniform R.C. line model definition syntax:

```
.MODEL mymodelname URC ( <Other_Model_Parameters…> )
```

Uniform R.C. line model definition parameters:

Parameter Name	Parameter Description	Units	Default
CPERL	Capacitance per unit length.	F/m	1×10^{-12}
FMAX	Maximum frequency of interest.	Hz	1×10^{9}
ISPERL	Saturation current per length.	A/m	—
K	Propagation constant.	—	1.5
RPERL	Resistance per unit length.	Ω/m	1000
RSPERL	Diode resistance per length.	Ω/m	—

Example

```
U1 1 2 0 myURC
.model myURC URC(isperl=1e-9)
```

JFET

JFET instance declaration syntax:

```
Jxxx nodeDrain nodeGate nodeSource Model <area> <OFF>
<IC=vds0, vgs0>
```

JFET instance declaration parameters:

Parameter Name	Parameter Description
area	Area factor.
OFF	[FLAG] Device is initially off.

Parameter Name	Parameter Description
IC-VDS	Initial drain-to-source voltage.
IC-VGS	Initial gate-to-source voltage.
IC	Initial voltages. This is a two element vector alternate way of specifying IC-VDS, IC-VGS.
TEMP	Instance temperature.

N-channel JFET model definition syntax:

```
.MODEL mymodelname NJF ( <Other_Model_Parameters…> )
```

P-channel JFET model definition syntax:

```
.MODEL mymodelname PJF ( <Other_Model_Parameters…> )
```

JFET model definition parameters:

Parameter Name	Parameter Description	Units	Default
AF	Flicker noise exponent.	—	1.0
ALPHA	Ionizing coefficient.	1/V	0.0
B	Doping tail parameter.	—	1.0
BETA	Transconductance parameter.	$A/(V^2)$	1×10^{-4}
BETATCE	BETA exponential temperature coefficient.	%/ºC	0.0
CGD	Zero-bias gate-to-drain junction capacitance.	F	0.0
CGS	Zero-bias gate-to-source junction capacitance.	F	0.0
FC	Forward bias junction fit parameter.	—	0.0
IS	Gate junction saturation current.	A	1×10^{-14}
ISR	Gate junction recombination current parameter.	A	0.0
KF	Flicker noise coefficient.	—	0.0
LAMBDA	Channel length modulation parameter.	1/V	0.0
M	Gate junction grading coefficient.	—	0.5

Parameter Name	Parameter Description	Units	Default
N	Gate junction emission coefficient.	—	1.0
NR	Emission coefficient for ISR.	—	2.0
PB	Gate junction potential.	V	1.0
RD	Drain ohmic resistance.	Ω	0.0
RS	Source ohmic resistance.	Ω	0.0
TNOM	Temperature at which model parameters were measured.	ºC	—
T_ABS	Absolute temperature.	ºC	—
T_MEASURED	Temperature at which model parameters were measured.	ºC	—
T_REL_GLOBAL	Temperature delta relative to global temperature.	ºC	—
T_REL_LOCAL	Temperature delta relative to AKO model temperature.	ºC	—
VK	Ionization knee voltage.	V	0.0
VTO or VT0	Threshold voltage.	V	–2.0
VTOTC	VTO temperature coefficient.	V/ºC	0.0
XTI	IS temperature coefficient.	—	3.0

Example

```
J1 d g s myJFET
.model myJFET NJF(vto=1.3)
```

MESFET

MESFET instance declaration syntax:

```
Zxxx nodeDrain nodeGate nodeSource Model <area> <OFF>
<ICVDS=icvds> <ICVGS=icvgs>
```

MESFET instance declaration parameters:

Parameter Name	Parameter Description
area	Area factor.
OFF	[FLAG] Device is initially off.
ICVDS	Initial drain-to-source voltage.
ICVGS	Initial gate-to-source voltage.
TEMP	Instance temperature.

N-channel MESFET model definition syntax:

```
.MODEL mymodelname NMF ( <Other_Model_Parameters…> )
```

P-channel MESFET model definition syntax:

```
.MODEL mymodelname PMF ( <Other_Model_Parameters…> )
```

MESFET model definition parameters:

Parameter Name	Parameter Description	Units	Default
AF	Flicker noise exponent.	—	1.0
ALPHA	Saturation voltage parameter.	1/V	2.0
B	Doping tail parameter.	—	0.3
BETA	Transconductance parameter.	$A/(V^2)$	2.5×10^{-3}
BETATCE	BETA exponential temperature coefficient.	%/ºC	0.0
CDS	Drain-to-source junction capacitance.	F	0.0
CGD	Gate-to-drain junction capacitance.	F	0.0
CGS	Gate-to-source junction capacitance.	F	0.0
EG	Bandgap voltage.	eV	1.11
FC	Forward bias depletion capacitance coefficient.	—	0.5
IS	Junction saturation current.	A	1×10^{-14}
KF	Flicker noise coefficient.	—	0.0
LAMBDA	Channel length modulation parameter.	1/V	0.0

Parameter Name	Parameter Description	Units	Default
M	Gate junction grading coefficient.	—	0.5
N	Gate junction emission coefficient.	—	1.0
PB or VBI	Gate junction potential.	V	1.0
RD	Drain ohmic resistance.	Ω	0.0
RG	Gate ohmic resistance.	Ω	0.0
RS	Source ohmic resistance.	Ω	0.0
TAU	Conduction current delay time.	sec	0.0
TNOM	Temperature at which model parameters were measured.	ºC	—
TRD1	RD linear temperature coefficient.	1/ºC	0.0
TRG1	RG linear temperature coefficient.	1/ºC	0.0
TRS1	RS linear temperature coefficient.	1/ºC	0.0
T_ABS	Absolute temperature.	ºC	—
T_MEASURED	Temperature at which model parameters were measured.	ºC	—
T_REL_GLOBAL	Temperature delta relative to global temperature.	ºC	—
T_REL_LOCAL	Temperature delta relative to AKO model temperature.	ºC	—
VDELTA	Capacitance transition voltage.	V	0.2
VMAX	Capacitance limiting voltage.	V	0.5
VTO or VT0	Pinch-off voltage.	V	-2.0
VTOTC	VTO temperature coefficient.	V/ºC	0.0
XTI	IS temperature coefficient.	—	0.0

Example

```
Z1 d g s myGaSFET
.model myGaSFET NMF(vto=1.3)
```

Voltage-Controlled Switch

Voltage-Controlled switch instance declaration syntax:

```
Sxxx node_n+ node_n- nodeNC+ nodeNC- Model <ON/OFF>
```

Voltage-Controlled Switch instance declaration parameters:

Parameter Name	Parameter Description
ON	[FLAG] Switch initially closed.
OFF	[FLAG] Switch initially open.

Voltage-Controlled Switch model definition syntax:

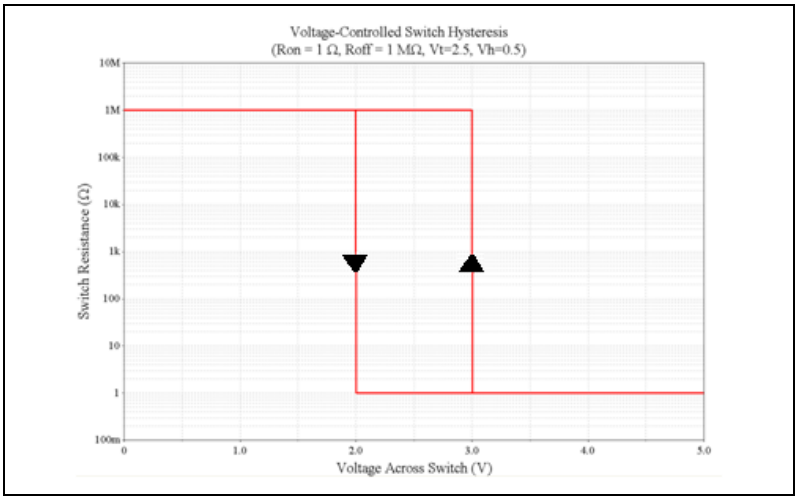
```
.MODEL mymodelname SW ( <Other_Model_Parameters…> )
```

Voltage-Controlled Switch model definition parameters:

Parameter Name	Parameter Description	Units	Default
ROFF	Resistance when open.	Ω	1/GMIN
RON	Resistance when closed.	Ω	1.0
VH	Hysteresis voltage.	V	0.0
VT	Threshold voltage.	V	0.0

Description

As illustrated below the voltage controlled switch changes abruptly between the ON and OFF states. Due to the hysteresis of the switch model the ON->OFF and OFF->ON transitions occur at VT+VH and VT-VH, respectively.



Example

```
S1 1 0 2 0 mySwitch
.model mySwitch SW(Ron=1 Roff=1meg Vt=2.5 vh=0.5)
```

Smooth Transition Voltage-Controlled Switch

Smooth transition Voltage-Controlled switch instance declaration syntax:

```
Sxxx node_n+ node_n- nodeNC+ nodeNC- Model
```

```
Sxxx node_n+ node_n- (nodeNC+, nodeNC-) Model
```

Smooth transition Voltage-Controlled Switch model definition syntax:

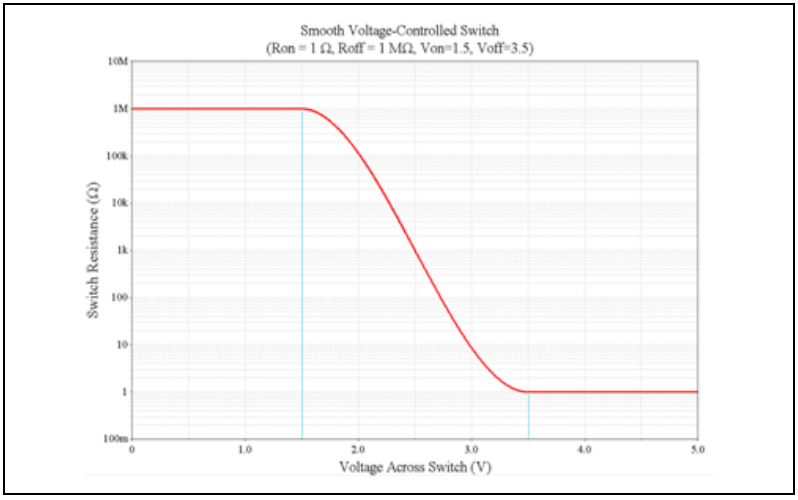
```
.MODEL mymodelname VSWITCH ( <Other_Model_Parameters…> )
```

Smooth transition Voltage-Controlled Switch model definition parameters:

Parameter Name	Parameter Description	Units	Default
ROFF	Resistance when open.	Ω	1×10^{12}
RON	Resistance when closed.	Ω	1.0
VOFF	Control OFF value.	V	0.0
VON	Control ON value.	V	1.0

Description

The switching characteristic of the smooth transition voltage-controlled switch are illustrated below. Since this device does not exhibit hysteresis the transition between the ON and OFF states follows the same characteristic curve.



Example

```
S1 1 0 2 0 mySwitch
.model mySwitch vswitch(Ron=1 Roff=1meg von=1.5
voff=3.5)
```

Current-Controlled Switch

Current-Controlled Switch instance declaration syntax:

```
Wxxx node_n+ node_n- Vname Model <ON/OFF>
```

Current-Controlled Switch instance declaration parameters:

Parameter Name	Parameter Description
Vname	Name of controlling voltage source.
ON	Switch initially closed.
OFF	Switch initially open.

Current-Controlled switch model definition syntax:

```
.MODEL mymodelname CSW ( <Other_Model_Parameters…> )
```

Current-Controlled Switch model (CSW) line parameters:

Parameter Name	Parameter Description	Units	Default
IH	Hysteresis current.	A	0.0
IT	Threshold current.	A	0.0
ROFF	Resistance when open.	Ω	1/GMIN
RON	Resistance when closed.	Ω	1.0

Description

The Current-Controlled switch follows a similar hysteresis curve as the Voltage-Controlled switch. Refer to the *Voltage-Controlled Switch* section for more information.

Example

```
S1 1 0 Vcontrol mySwitch
.model mySwitch CSW(Ron=0.1 Roff=1meg it=2 ih=0.1)
```

BJT

BJT device instance declaration syntax:

```
Qxxx nodeCollector nodeBase nodeEmmiter <nodeBody> Model
<area> <OFF> <IC=VBE0, VCE0>
```

BJT instance declaration parameters:

Parameter Name	Parameter Description
area	Area factor.
OFF	[FLAG] Device is initially off.
ICVBE	Initial base-emitter voltage.
ICVCE	Initial collector-emitter voltage.
IC	Initial voltages. This is a two element vector alternate way of specifying ICVBE, ICVCE.
SENS_AREA	[FLAG] Flag to request sensitivity with respect to area.
TEMP	Instance temperature.

BJT device NPN model definition syntax:

```
.MODEL mymodelname NPN (<Other_Model_Parameters…> )
```

BJT device PNP model definition syntax:

```
.MODEL mymodelname PNP (<Other_Model_Parameters…> )
```

BJT device LPNP model definition syntax:

```
.MODEL mymodelname LPNP (<Other_Model_Parameters…> )
```

BJT model definition parameters:

Parameter Name	Parameter Description	Units	Default
AF	Flicker noise exponent.	—	1.0
BF	Ideal maximum forward beta.	—	100.0
BR	Ideal maximum reverse beta.	—	1.0
CJC	Base-collector zero bias depletion capacitance.	F	0.0
CJE	Base-emitter zero bias depletion capacitance.	F	0.0

Parameter Name	Parameter Description	Units	Default
CJS or CCS	Collector-substrate zero bias depletion capacitance.	F	0.0
CN	Quasi-saturation temperature coefficient for hole mobility.	—	2.42 NPN 2.20 PNP
D	Quasi-saturation temperature coefficient for scattering-limited hole carrier velocity.	—	0.87 NPN 0.52 PNP
EG	Bandgap voltage.	eV	1.11
FC	Forward bias depletion capacitance coefficient.	—	0.5
GAMMA	Epitaxial region doping factor.	—	1×10^{-11}
IKF or IK	Corner for forward beta high current roll-off.	A	infinite
IKR	Corner for reverse beta high current roll-off.	A	infinite
IRB	Current at which base resistance is (RB+RBM)/2.	A	infinite
IS	Transport saturation current.	A	1×10^{-16}
ISC	Base-collector leakage saturation current.	A	0.0
ISE	Base-emitter leakage saturation current.	A	0.0
ISS	Substrate junction saturation current.	A	0.0
ITF	High current dependence of TF.	A	0.0
KF	Flicker noise coefficient.	—	0.0
MJC or MC	Base-collector junction grading coefficient.	—	0.33
MJE or ME	Base-emitter junction grading coefficient.	—	0.33
MJS or MS	Substrate junction grading coefficient.	—	0.0
NC	Base-collector leakage emission coefficient.	—	2.0
NE	Base-emitter leakage emission coefficient.	—	1.5
NF	Forward current emission coefficient.	—	1.0
NK	High-current roll-off coefficient.	—	0.5
NR	Reverse current emission coefficient.	—	1.0

Parameter Name	Parameter Description	Units	Default
NS	Substrate junction emission coefficient.	—	1.0
PTF	Excess phase at 1/(2πTF) Hz.	º	0.0
QCO	Epitaxial region charge factor.	C	0.0
QUASIMOD	Quasi-saturation model flag for GAMMA, RCO, and VO temperature dependence: = 1 perform temperature adjustment = 0 don't perform temperature adjustment	—	0
RB	Zero bias base resistance.	Ω	0.0
RBM	Minimum base resistance.	Ω	RB
RC	Collector resistance.	Ω	0.0
RCO	Epitaxial region resistance.	Ω	0.0
RE	Emitter resistance.	Ω	0.0
TF	Ideal forward transient time.	s	0.0
TNOM	Temperature at which model parameters were measured.	ºC	—
TR	Ideal reverse transit time.	s	0.0
TRB1	RB linear temperature coefficient.	1/ºC	0.0
TRB2	RB quadratic temperature coefficient.	$1/(ºC)^2$	0.0
TRC1	RC linear temperature coefficient.	1/ºC	0.0
TRC2	RC quadratic temperature coefficient.	$1/(ºC)^2$	0.0
TRE1	RE linear temperature coefficient.	1/ºC	0.0
TRE2	RE quadratic temperature coefficient.	$1/(ºC)^2$	0.0
TRM1	RBM linear temperature coefficient.	1/ºC	0.0
TRM2	RBM quadratic temperature coefficient.	$1/(ºC)^2$	0.0
T_ABS	Absolute temperature.	ºC	—
T_MEASURED	Temperature at which model parameters were measured.	ºC	—

Parameter Name	Parameter Description	Units	Default
T_REL_GLOBAL	Temperature delta relative to global temperature.	ºC	—
T_REL_LOCAL	Temperature delta relative to AKO model temperature.	ºC	—
VAF or VA	Forward Early voltage.	V	infinite
VAR or VB	Reverse Early voltage.	V	infinite
VG	Quasi-saturation extrapolated bandgap voltage at 0K.	V	1.206
VJC or PC	Base-collector built in potential.	V	0.75
VJE or PE	Base-emitter built in potential.	V	0.75
VJS or PS	Substrate junction built in potential.	V	0.75
VO	Carrier mobility knee voltage.	V	10.0
VTF	Voltage giving VBC dependence of TF.	V	infinite
XCJC	Fraction of base-collector capacitance connected to internal base.	—	1.0
XCJC2	Fraction of base-collector capacitance connected to internal base.	—	1.0
XCJS	Fraction of substrate-collector capacitance connected to internal collector.	—	1.0
XTB	Forward and reverse beta temperature exponent.	—	0.0
XTF	Coefficient for bias dependence of TF.	—	0.0
XTI	IS temperature effect exponent.	—	3.0

Additional Notes

Multisim uses the Gummel-Poon BJT model.

Example

```
Q1 e b c 0 myBJT
.model myBJT NPN(vto=1.3)
```

MOSFET

MOSFET device instance declaration syntax:

```
Mxxx nodeDrain nodeGate nodeSource nodeBulk Model
<<L=>l> <<W=>w> <AD=ad> <AS=as> <PD=pd> <PS=ps>
+ <NRD=nrd> <NRS=nrs> <M=m> <OFF> <IC=vds <, vgs <,
vbs>>>

Mxxx nodeDrain nodeGate nodeSource nodeBulk Model
<<L=>l> <<W=>w> <AD=ad> <AS=as> <PD=pd> <PS=ps>
+ <NRD=nrd> <NRS=nrs> <M=m> <OFF> <VDS=vds> <VGS=vgs>
<VBS=vbs>
```

Basic MOSFET instance declaration parameters:

Parameter Name	Parameter Description
L	Length.
W	Width.
AD	Drain area.
AS	Source area.
PD	Drain perimeter.
PS	Source perimeter.
NRD	Number of squares in drain.
NRS	Number of squares in source.
M	Device multiplicity factor.
OFF	[FLAG] Device is initially off.
ICVDS	Initial drain-to-source voltage.
ICVGS	Initial gate-to-source voltage.
ICVBS	Initial bulk-to-source voltage.
IC	Initial voltages. This is a three element vector alternate way of specifying ICVDS, ICVGS, ICVBS.
SENS_L	[FLAG] flag to request sensitivity with respect to length.

Parameter Name	Parameter Description
SENS_W	[FLAG] flag to request sensitivity with respect to width.
TEMP	Instance temperature.

MOSFET device NMOS simulation model definition syntax:

```
.MODEL mymodelname NMOS (<LEVEL=level>
<Other_Model_Parameters…> )
```

MOSFET device PMOS simulation model definition syntax:

```
.MODEL mymodelname PMOS (<LEVEL=level>
<Other_Model_Parameters…> )
```

The LEVEL parameter is used to select the appropriate MOSFET simulation model. Multisim provides eight different MOSFET models, which are described below:

Level Value	Description
1 or MOS1	Shichman-Hodges model (DEFAULT model).
2 or MOS2	More complex model than LEVEL 1 based on actual device physics.
3 or MOS3	Semi-empirical model good for simulating short channel effects.
4 or BSIM1	BSIM1.
5 or BSIM2	BSIM2.
6 or MOS6	N-th power law MOSFET model.
8 or BSIM3	BSIM3 (version 3v3).
14 or BSIM	BSIM4 (version 4v5).

Depending on the Level value, different parameters for both instance declarations and model definitions are available.

MOSFET model definition parameters for MOS1 and MOS2:

Parameter Name	Parameter Description	Units	Default
AF	Flicker noise exponent.	—	1.0
CBD	Bulk-to-drain junction capacitance.	F	0.0
CBS	Bulk-to-source junction capacitance.	F	0.0
CGBO	Gate-to-bulk overlap capacitance.	F/m	0.0
CGDO	Gate-to-drain overlap capacitance.	F/m	0.0
CGSO	Gate-to-source overlap capacitance.	F/m	0.0
CJ	Zero-bias bulk junction bottom capacitance per area.	$F/(m^2)$	0.0
CJSW	Zero-bias bulk junction sidewall capacitance per length.	F/m	0.0
FC	Forward bias depletion capacitance coefficient.	—	0.5
GAMMA	Bulk threshold parameter.	$V^{1/2}$	0.0
GDSNOI	Channel shot noise coefficient (use when NLEV=3).	—	1.0
IS	Bulk junction saturation current.	A	1×10^{-14}
JS	Bulk junction saturation current density.	$A/(m^2)$	0.0
JSSW	Bulk junction saturation sidewall current per length.	A/m	0.0
KF	Flicker noise coefficient.	—	0.0
KP	Transconductance parameter.	$A/(V^2)$	2×10^{-5}
L	Length.	m	DEFW
LAMBDA	Channel length modulation.	1/V	0.0
LD	Lateral diffusion (length).	m	0.0
MJ	Bulk junction bottom grading coefficient.	—	0.5
MJSW	Bulk junction sidewall grading coefficient.	—	0.33
N	Bulk junction emission coefficient.	—	1.0

Parameter Name	Parameter Description	Units	Default
NLEV	Noise equation selector.	—	2
NSS	Surface state density.	$1/(cm^2)$	0.0
NSUB	Substrate doping.	$1/(cm^2)$	0.0
PB	Bulk junction potential.	V	0.8
PBSW	Bulk junction sidewall potential.	V	PB
PHI	Surface potential.	V	0.6
RB	Bulk ohmic resistance.	Ω	0.0
RD	Drain ohmic resistance.	Ω	0.0
RDS	Drain-to-source shunt resistance.	Ω	infinite
RG	Gate ohmic resistance.	Ω	0.0
RS	Source ohmic resistance.	Ω	0.0
RSH	Drain and source diffusion sheet resistance.	Ω/sq.	0.0
TT	Bulk junction transit time.	sec	0.0
TNOM	Temperature at which model parameters were measured.	ºC	—
T_ABS	Absolute temperature.	ºC	—
T_MEASURED	Temperature at which model parameters were measured.	ºC	—
T_REL_GLOBAL	Temperature delta relative to global temperature.	ºC	—
T_REL_LOCAL	Temperature delta relative to AKO model temperature.	ºC	—
TOX	Oxide thickness.	m	Refer to *Additional Notes*
TPG	Type of gate material: +1 = opposite to substrate –1 = same as substrate 0 = aluminum.	—	1

Parameter Name	Parameter Description	Units	Default
U0	Surface mobility.	cm^2/Vs	600
VTO	Threshold voltage.	V	0.0
W	Width.	m	DEFL
WD	Lateral diffusion (width).	m	0.0

Additional MOSFET model definition parameters for MOS3:

Parameter Name	Parameter Description	Units	Default
AF	Flicker noise exponent.	—	1.0
ALPHA	Alpha.	—	0.0
DELTA	Width effect on threshold.	—	0.0
ETA	Static feedback.	—	0.0
KAPPA	Saturation field factor.	—	0.2
KF	Flicker noise coefficient.	—	0.0
L	Length.	m	DEFW
NFS	Fast surface state density.	$1/(cm^2)$	0.0
RB	Bulk ohmic resistance.	Ω	0.0
RG	Gate ohmic resistance.	Ω	0.0
THETA	Mobility modulation.	1/V	0.0
VMAX	Maximum carrier drift velocity.	m/s	0.0
W	Width.	m	DEFL
XD	Depletion layer width.	—	0.0
XJ	Metallurgical junction depth.	m	0.0

Additional MOSFET model definition parameters for MOS6:

Parameter Name	Parameter Description	Units	Default
GAMMA1	Bulk threshold parameter 1.	$V^{1/2}$	0.0
KC	Saturation current factor.	—	5×10^{-5}
KV	Saturation voltage factor.	—	2.0
LAMBDA0	Channel length modulation parameter 0.	—	0.0
LAMBDA1	Channel length modulation parameter 1.	1/V	0.0
NC	Saturation current coefficient.	—	1.0
NV	Saturation voltage coefficient.	—	0.5
NVTH	Threshold voltage coefficient.	—	0.5
PS	Saturation current modification parameter.	—	0.0
SIGMA	Static feedback effect parameter.	—	0.0

Additional Notes

- The level parameter (level) must be a numerical constant— it may not contain parameters or expressions.
- The oxide thickness parameter (TOX) has default value of 1x10-7 for LEVEL 2 and LEVEL 3 models. For the LEVEL 1 the default value is unspecified, and if TOX is not provided the model does not use the process parameters, for example, TOX, NSUB, COX, UO).
- Please consult external documentation at `http://www-device.eecs.berkeley.edu/~bsim` for additional details on BSIM models.

Example

```
M1 d g s s myMOS w=200u l=100u m=3
.model myMOS NMOS(vto=1.3)
```

Independent Voltage Source

Independent Voltage Source instance declaration syntax:

```
Vxxx node1 node2 <<DC> dc_mag> <AC <ac_mag <ac_phase>>
<source_type(source_params)>
```

Independent Voltage Source instance declaration parameters:

Parameter Name	Parameter Description
DC	DC voltage identifier.
AC	AC analysis input voltage identifier.
dc_mag	DC magnitude voltage value.
ac_mag	AC signal magnitude value.
ac_phase	AC signal phase value.
source_type	Type of source waveform (see table below).
source_params	Parameters for source waveform.

The types of voltage source waveforms allowed by Multisim are:

source_type	Description
PULSE	Pulse source. Refer to the *Pulse Source* section for information.
SIN	Sinusoidal source. Refer to the *Sinusoidal Source* section for information.
SFFM	Single frequency FM source. Refer to the *Single Frequency FM Source* section for information.
EXP	Exponential source. Refer to the *Exponential Source* section for information.
PWL	Piecewise linear source. Refer to the *Piecewise Linear Source* section for information.
PWLREPEAT	Repeating piecewise linear source. Refer to the *Piecewise Linear Source* section for information.
PWLFILE	PWL source generated using data stored in a file. Refer to the *Piecewise Linear File Source* section for information.
PWLFILEREPEAT	Repeating PWL source generated using data stored in a file. Refer to the *Piecewise Linear File Source* section for information.

source_type	Description
XM	Agilent 33120A Arbitrary Waveform Generator modulation. Refer to the *Modulation Source* section for information.
XAM	Agilent 33120A Arbitrary Waveform Generator AM modulation. Refer to the *AM Modulation Source* section for information.
XFM	Agilent 33120A Arbitrary Waveform Generator FM modulation. Refer to the *FM Modulation Source* section for information.
XFSK	Agilent 33120A Arbitrary Waveform Generator FSK modulation. Refer to the *FSK Modulation Source* section for information.
XBST	Agilent 33120A Arbitrary Waveform Generator Burst modulation. Refer to the *BST Modulation Source* section for information.
XSWP	Agilent 33120A Arbitrary Waveform Generator Sweep modulation. Refer to the *SWP Modulation Source* section for information.
XNOISE	Agilent 33120A Arbitrary Waveform Generator Noise modulation. Refer to the *Noise Source* section for information.
XARB	Agilent 33120A Arbitrary Waveform Generator ARB modulation. Refer to the *XARB Source* for information.

Description

An independent voltage source is an ideal voltage source, fully driving the voltage between its nodes. Every voltage source statement has three optional portions describing its output behavior: DC portion which describes its behavior in DC analysis, AC portion which describes its behavior in AC analysis, and source_type which describes its behavior in Transient Analysis.

Additional Notes

- In DC analysis, if the DC portion is not specified, but the source_type portion is specified, the output voltage is that of the source_type at time=0.0.
- In transient analysis, if the source_type portion is not specified, but the DC portion is specified, the output voltage for the duration of the analysis is the DC value.
- If neither the DC portion or the source_type portion are specified, the output is zero volts.
- Sources have special parameters that are enclosed in brackets; these brackets are optional.
- Due to the ambiguity of some source parameter names (SIN, for example) expressions used in source statements should always have curly brackets, {}, around them.
- The files used in PWLFILE and PWLFILEREPEAT should be plain text, with one comma-separated pair of numbers on every line of the file. For more details on the file format, refer to the *Piecewise Linear Source* section.

Examples

```
V1 in 0 10

*1khz, 10V, 50% duty cycle voltage signal
V1 in ref pulse (10 0 0 1n 1n 0.5m 1m)

*5V DC source with a 1V voltage for AC analysis
V1 1 0 ac 1 dc 5
```

Pulse Source

Pulse source instance declaration syntax:

```
PULSE( vi vp <td <tr <tf <pw <per>>>>> )
```

PULSE instance declaration parameters:

Parameter Name	Parameter Description
vi	Initial value (voltage/current).
vp	Pulsed value voltage/current).
td	Time delay.

Parameter Name	Parameter Description
tr	Rise time.
tf	Fall time.
pw	Pulse width.
per	Period.

Sinusoidal Source

Sinusoidal source instance declaration syntax:

```
SIN( vo va <freq <td <theta>>> )
```

SIN instance declaration parameters:

Parameter Name	Parameter Description
vo	Offset value.
va	Peak value.
freq	Frequency.
td	Delay.
theta	Phase.

Single Frequency FM Source

Single frequency FM source instance declaration syntax:

```
SFFM( vo va <fc <mdi <fs>>> )
```

SFFM instance declaration parameters:

Parameter Name	Parameter Description
vo	Offset value.
va	Peak value.
fc	Carrier frequency.
mdi	Modulation index.
fs	Signal frequency.

Exponential Source

Exponential source instance declaration syntax:

```
EXP( vi vp <td1 tau1 td2 <tau2>> )
```

EXP instance declaration parameters:

Parameter Name	Parameter Description
vi	Initial value.
vp	Peak value.
td1	Rise/fall delay time.
tau1	Rise/fall time constant.
td2	Fall/rise delay time.
tau2	Fall/rise time constant.

Piecewise Linear Source

Piecewise linear source instance declaration syntax:

```
PWL( t1 v1 <t2 v2 <t3 v3 <...>>> )
```

```
PWL( t1, v1 ) <( t2, v2 ) <( t3, v3 ) <...>>> )
```

Repeating piecewise linear source instance declaration syntax:

```
PWLREPEAT( t1 v1 <t2 v2 <t3 v3 <...>>> )
```

```
PWLREPEAT( t1, v1 ) <( t2, v2 ) <( t3, v3 ) <...>>> )
```

PWL and PWLREPEAT instance declaration parameters:

Parameter Name	Parameter Description
t*n* where n=0,1,2…	Time component of data point *n*.
v*n* where n=0,1,2…	Voltage component of data point *n*.

Piecewise Linear File Source

File-based piecewise linear source instance declaration syntax:

```
PWLFILE( filename_string )
```

Repeating File-based piecewise linear source instance declaration syntax:

```
PWLFILEREPEAT( filename_string )
```

PWLFILE and PWLFILEREPEAT instance declaration parameters:

Parameter Name	Parameter Description
filename_string	A fully qualified filename in quotes (for example, "c:\mypath\myfile.ext").

Modulation Source

This source models the modulation available on the Agilent 33120A Arbitrary Waveform Generator.

XM instrument voltage source instance declaration syntax:

```
XM( vo va fc td dt cwid )
```

Parameter Name	Parameter Description
vo	Offset value.
va	Amplitude value.
fc	Carrier frequency.
td	Time delay.
dt	Duty value for square wave.
cwid	Carrier wave ID: 0=sine, 1=square, 2=triangle, 3=ramp.

AM Modulation Source

This source models the AM modulation available on the Agilent 33120A Arbitrary Waveform Generator.

AM modulation voltage source instance declaration syntax:

```
XAM( vo va fc td dt depth fs cwid wsid )
```

Parameter Name	Parameter Description
vo	Offset value.
va	Amplitude value.
fc	Carrier frequency.
td	Time delay.
dt	Duty value for square wave.
depth	Modulation depth.
fs	Signal frequency.
cwid	Carrier wave ID: 0=sine, 1=square, 2=triangle, 3=ramp.
wsid	Signal wave ID: 0=sine, 1=square, 2=triangle, 3=ramp.

FM Modulation Source

This source models the FM modulation available on the Agilent 33120A Arbitrary Waveform Generator.

FM modulation voltage source instance declaration syntax:

```
XFM( vo va fc mdi fs cwid wsid )
```

Parameter Name	Parameter Description
vo	Offset value.
va	Amplitude value.
fc	Carrier frequency.
mdi	Modulation index.

Parameter Name	Parameter Description
fs	Signal frequency.
cwid	Carrier wave ID: 0=sine, 1=square, 2=triangle, 3=ramp.
wsid	Signal wave ID: 0=sine, 1=square, 2=triangle, 3=ramp.

FSK Modulation Source

This source models the FSK modulation available on the Agilent 33120A Arbitrary Waveform Generator.

FSK modulation voltage source instance declaration syntax:

```
XFSK( vo va fc td dt fh fskr )
```

Parameter Name	Parameter Description
vo	Offset value.
va	Amplitude value.
fc	Carrier frequency.
td	Time delay.
dt	Duty value for square wave.
fh	Hop frequency.
fskr	FSK rate.

BST Modulation Source

This source models the Burst modulation available on the Agilent 33120A Arbitrary Waveform Generator.

BST modulation voltage source instance declaration syntax:

```
XBST( vo va fc td dt bstr nc ph cwid tm te )
```

Parameter Name	Parameter Description
vo	Offset value.
va	Amplitude value.
fc	Carrier frequency.
td	Time delay.
dt	Duty value for square wave.
bstr	BST rate.
nc	BST counter.
ph	Phase (in degrees).
cwid	Carrier wave ID: 0=sine, 1=square, 2=triangle, 3=ramp.
tm	Trigger mode: 0=auto, 1=single.
te	Trigger enable: 0=disabled, 1=enabled.

SWP Modulation Source

This source models the Sweep modulation available on the Agilent 33120A Arbitrary Waveform Generator.

SWP modulation voltage source instance declaration syntax:

```
XSWP( vo va td dt fb fe ts sm cwid tm te)
```

Parameter Name	Parameter Description
vo	Offset value.
va	Amplitude value.
td	Time delay.
dt	Duty value for square wave.
fb	Start frequency.
fe	End frequency.
ts	Sweep time.

Parameter Name	Parameter Description
sm	Sweep mode: 1=linear, 0=log.
cwid	Signal wave ID: 0=sine, 1=square, 2=triangle, 3=ramp.
tm	Trigger mode: 0=auto, 1=single.
te	Trigger enable: 0=disabled, 1=enabled.

Noise Source

This source models the Noise modulation available on the Agilent 33120A Arbitrary Waveform Generator.

NOISE modulation voltage source instance declaration syntax:

```
XNOISE( vo va fc bf)
```

Parameter Name	Parameter Description
vo	Offset value.
va	Amplitude value.
fc	Carrier frequency.
bf	Bandwidth frequency.

XARB Source

This source models the ARB modulation available on the Agilent 33120A Arbitrary Waveform Generator.

XARB voltage source instance declaration syntax:

```
XARB( tt td t0 v0 t1 v1 <t2 v2 <...>>)
```

Parameter Name	Parameter Description
tt	Period time (1/F).
td	Time delay.
tn where n=0,1,2…	Time component of data point *n*.
vn where n=0,1,2…	Voltage component of data point *n*.

Independent Current Source

Independent Current Source instance declaration syntax:

```
Ixxx node1 node2 <<DC> dc_mag> <AC <ac_mag <ac_phase>>
<source_type(source_params)>
```

Independent Current Source instance declaration parameters:

Parameter Name	Parameter Description
DC	DC current identifier.
AC	AC analysis input current identifier.
dc_mag	DC magnitude current value.
ac_mag	AC signal magnitude value.
ac_phase	AC signal phase value.
source_type	Type of source waveform (see table below).
source_params	Parameters for source waveform.

The types of current source waveforms allowed by Multisim are:

source_type	Description
PULSE	Pulse source. Refer to the *Pulse Source* section for information.
SIN	Sinusoidal source. Refer to the *Sinusoidal Source* section for information.
SFFM	Single frequency FM source. Refer to the *Single Frequency FM Source* section for information.
EXP	Exponential source. Refer to the *Exponential Source* section for information.
PWL	Piecewise linear source. Refer to the *Piecewise Linear Source* section for information.
PWLREPEAT	Repeating piecewise linear source. Refer to the *Piecewise Linear Source* section for information.

source_type	Description
PWLFILE	PWL source generated using data stored in a file. Refer to the *Piecewise Linear File Source* section for information.
PWLFILEREPEAT	Repeating PWL source generated using data stored in a file. Refer to the *Piecewise Linear File Source* section for information.

Description

An independent current source is an ideal current source, fully driving the current in its branch. Every current source statement has three optional portions describing its output behavior: DC portion which describes its behavior in DC analysis, AC portion which describes its behavior in AC analysis, and source_type which describes its behavior in Transient Analysis.

Additional Notes

- In DC analysis, if the DC portion is not specified, but the source_type portion is specified, the output voltage is that of the `source_type` at time=0.0.
- In transient analysis, if the `source_type` portion is not specified, but the DC portion is specified, the output voltage for the duration of the analysis is the DC value.
- If neither the DC portion or the `source_type` portion are specified, the output is zero volts.
- Sources have special parameters that are enclosed in brackets; these brackets are optional.
- Due to the ambiguity of some source parameter names (SIN, for example) expressions used in source statements should always have curly brackets, {}, around them.
- The files used in PWLFILE and PWLFILEREPEAT should be plain text, with one comma-separated pair of numbers on every line of the file.

Examples

```
I1 0 in 10

*1khz, 5Apk current source
I1 0 in sin (0 5 1k)
```

```
*5A DC source with a 1A for AC analysis
I1 0 1 ac 1 dc 5
```

Arbitrary Source

Arbitrary Source instance declaration syntax:

```
Bxxx o_node_p o_node_n V = expression
Bxxx o_node_p o_node_n I = expression

Exxx o_node_p o_node_n VALUE = expression
Exxx o_node_p o_node_n TABLE(expression) < = > (x1,y1)
(x2,y2) < (x3,y3) <...>>

Gxxx o_node_p o_node_n VALUE = expression
Gxxx o_node_p o_node_n TABLE(expression) < = > (x1,y1)
(x2,y2) < (x3,y3) <...>>
```

Arbitrary Source instance declaration terminals:

Parameter Name	Parameter Description
o_node_p	Output node +
o_node_n	Output node –

Arbitrary Source instance declaration parameters:

Parameter Name	Parameter Description
expression	A mathematical expression for the output variable.
x*N* where N=0,1,2…	*N*th x value for a TABLE interpolation.
y*N* where N=0,1,2…	*N*th y value for a TABLE interpolation.

Description

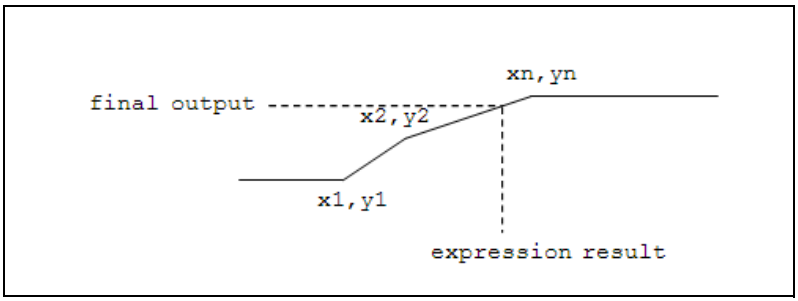
The arbitrary source generates an output voltage or current that is equal to the result of an arbitrary mathematical expression operating on any number of circuit variables. This ability makes it much more powerful than the linear and polynomial controlled sources.

The B-source with the declaration "Bxxx o_node_p o_node_n V = expression" is a controlled voltage source and is functionality equivalent

to the E-source with the declaration "Exxx o_node_p o_node_n VALUE = expression"

The B-source with the declaration "Bxxx o_node_p o_node_n I = expression" is a controlled current source and is functionality equivalent to the G-source with the declaration "Gxxx o_node_p o_node_n VALUE = expression"

The E and G sources based on TABLE perform an additional step of mapping the result of expression to the piece-wise-linear function described by co-ordinates (x1,y1) (x2,y2)… (xn,yn).



Example

```
E1 5 0 TABLE(V(1)**2) (-5,-5)(5,5)
B99 5 0 v={limit(log(v(5)),-100,10)}
G1 5 0 TABLE(V(1)**2) (-5,-5)(5,5)
B99 5 0 I={max(v(5),v(4))}
```

Voltage Controlled Voltage Source

Linear Voltage Controlled Voltage Source instance declaration syntax:

```
Exxx o_node_p o_node_n c_node0_p c_node0_n gain
```

```
Exxx o_node_p o_node_n < POLY(ndim) > c_node0_p
c_node0_n < c_node1_p c_node1_n <...>> vo p0 + <p1 <p2
<...>>>
```

Voltage Controlled Voltage Source instance declaration terminals:

Parameter Name	Parameter Description
o_node_p	Output node +.
o_node_n	Output node –.

Parameter Name	Parameter Description
c_node*N*_p where N=0,1,2…	*N*th controlling node +.
c_node*N*_n where N=0,1,2…	*N*th controlling node –.

Voltage Controlled Voltage Source instance declaration parameters:

Parameter Name	Parameter Description
gain	Voltage gain (same as p0)
ndim	The number of pairs of controlling nodes. This must be a simple integer, not an expression. Refer to the *Polynomial Specifications* section for more information.
vo	Voltage offset
p*N* where *N*=0,1,2…	*N*th polynomial term. Refer to the *Polynomial Specifications* section for more information.

Polynomial Specifications

In polynomial controlled sources the ndim value provided is the number of controlling inputs, which are either pairs of nodes or names of current sources. The number of inputs must match the ndim value, however the number of polynomial terms is independent.

There can be any number of polynomial terms which are used to generate an expression of the following form:

For POLY(1):

```
p0*a + p1*a*a + p2*a*a*a + p3*a*a*a*a + ...
```

where a is the value of V(c_node0_p, c_node0_n) or I(c_isrc0)

For POLY(2):

```
p0*a + p1*b + p2*a*a + p3*a*b + p4*b*b + p5*a*a*a +
p6*a*a*b + p7*a*b*b + p8*b*b*b + p9*a*a*a*a +
p10*a*a*a*b +...
```

where a is the value of V(c_node0_p, c_node0_n) or I(c_isrc0) and b is the value of V(c_node1_p, c_node1_n) or I(c_isrc1)

For POLY(3):

```
p0*a + p1*b + p2*c + p3*a*a + p4*a*b + p5*a*c + p6*b*b
+ p7*b*c + p8*c*c + p9*a*a*a + p10*a*a*b + p11*a*a*c
+ p12*a*b*b + p13*a*b*c + p14*a*c*c + p15*b*b*b +
p16*b*b*c + p17*b*c*c + p18*c*c*c + p19*a*a*a*a +
....
```

where a is the value of V(c_node0_p, c_node0_n) or I(c_isrc0)

and b is the value of V(c_node1_p, c_node1_n) or I(c_isrc1)

and c is the value of V(c_node2_p, c_node2_n) or I(c_isrc2)

etc…

Additional Notes

- The dimension parameter of the polynomial source (ndim) must be a numerical constant—it may not contain parameters or expressions.

Example

```
E1 ampout 0 ampin 0 1e6
E1 out ref poly(2) node1 0 input+ input- 0.5 1 1
```

Current Controlled Voltage Source

Current Controlled Voltage Source instance declaration sybtax:

```
Hxxx o_node_p o_node_n Vname1 gain

Hxxx o_node_p o_node_n < POLY(ndim) > Vname1 < Vname2
<...>> vo p0 <p1 <p2 <...>>>
```

Current Controlled Voltage Source instance declaration terminals:

Parameter Name	Parameter Description
o_node_p	Output node +.
o_node_n	Output node –.
c_node*N*_p where *N*=0,1,2…	*N*th controlling node +.
c_node*N*_n where *N*=0,1,2…	*N*th controlling node –.
Vname*N* where *N*=0,1,2…	*N*th controlling current source name.

Current Controlled Voltage Source instance declaration parameters:

Parameter Name	Parameter Description
gain	Voltage gain (same as p0).
ndim	The number of pairs of controlling nodes. This must be a simple integer. Refer to the *Polynomial Specifications* section for more information.
vo	Voltage offset.
p*N* where *N*=0,1,2…	Nth polynomial term. Refer to the *Polynomial Specifications* section for more information.

Additional Notes

- The dimension parameter of the polynomial source (ndim) must be a numerical constant - it may not contain parameters or expressions.

Example

```
H1 out 0 Vsense 1k
```

Voltage Controlled Current Source

Voltage Controlled Current Source instance declaration syntax:

```
Gxxx o_node_p o_node_n c_node0_p c_node0_n gain

Gxxx o_node_p o_node_n < POLY(ndim) > c_node0_p
c_node0_n < c_node1_p c_node1_n <...>> vo p0
+ <p1 <p2 <...>>>
```

Voltage Controlled Current Source instance declaration terminals:

Parameter Name	Parameter Description
o_node_p	Output node +.
o_node_n	Output node –.
c_node*N*_p where *N*=0,1,2…	*N*th controlling node +.
c_node*N*_n where *N*=0,1,2…	*N*th controlling node –.

Voltage Controlled Voltage Source instance declaration parameters:

Parameter Name	Parameter Description
gain	Voltage gain (same as p0).
ndim	The number of pairs of controlling nodes. This must be a simple integer, not an expression. Refer to the *Polynomial Specifications* section for more information.
vo	Voltage offset.
p*N* where *N*=0,1,2…	Nth polynomial term. Refer to the *Polynomial Specifications* section for more information.

Additional Notes

- The dimension parameter of the polynomial source (ndim) must be a numerical constant—it may not contain parameters or expressions.

Example

```
G1 0 out 1 2 100k
G1 out ref poly(2) node_a 0 node_b 0 0.5 1 1 2 2
```

Current Controlled Current Source

Current Controlled Current Source instance declaration syntax:

```
Fxxx o_node_p o_node_n Vname1 gain

Fxxx o_node_p o_node_n < POLY(ndim) > Vname1 < Vname2
<...>> vo p0 <p1 <p2 <...>>>
```

Current Controlled Current Source instance declaration terminals:

Parameter Name	Parameter Description
o_node_p	Output node +.
o_node_n	Output node –.
c_node*N*_p where *N*=0,1,2…	*N*th controlling node +.

Parameter Name	Parameter Description
c_node*N*_n where *N*=0,1,2…	*N*th controlling node –.
Vname*N* where *N*=0,1,2…	*N*th controlling current source name.

Current Controlled Voltage Source instance declaration parameters:

Parameter Name	Parameter Description
gain	Voltage gain (same as p0).
ndim	The number of pairs of controlling nodes. This must be a simple integer, not an expression. Refer to the *Polynomial Specifications* section for more information.
vo	Voltage offset.
p*N* where *N*=0,1,2…	Nth polynomial term. Refer to the *Polynomial Specifications* section for more information.

Additional Notes

- The dimension parameter of the polynomial source (ndim) must be a numerical constant—it may not contain parameters or expressions.

Example

```
F1 0 out V1 10
```

Temperature Parameters

Global Temperature Parameters

In the **Custom Analysis Options** dialog, two global temperature parameters, TEMP and TNOM, can be specified. The TEMP parameter is used to define the simulation temperature, and the TNOM parameter defines the temperature at which the model parameters were measured. Both parameters have a default value of 27° C.

Instance/Model Temperature Parameters

The TNOM and T_MEASURED model parameters override the global TNOM parameter.

Example:

```
.model Res R (TNOM=30)
```

The TEMP instance parameter overrides the global TEMP parameter.

Example:

```
.model Res R (TEMP=75)
```

The following Cadence® PSpice® temperature parameters also override the global TEMP parameter:

Parameter	Device Simulation Temperature	Description
T_ABS	T_ABS	Absolute simulation temperature. Example: `.MODEL PNP_MOS(ABS=55 T_MEASURED=0)`
T_REL_GLOBAL	<global temperature> + T_REL_GLOBAL	Temperature delta relative to the global simulation temperature. Example: `.MODEL PNP_MOS (T_REL_GLOBAL=-17)`

T_REL_LOCAL	<T_ABS in AKO model> + T_REL_LOCAL	Temperature delta relaive to T_ABS specified inside the AKO model.
		Example:
		`.MODEL PNP MOD1 T_ABS=60`
		`.MODEL MOD2 AKO:MOD1 PNP T_REL_LOCAL=10`

The Cadence® PSpice® temperature parameters (T_MEASURED, T_ABS, T_REL_LOCAL, T_REL_GLOBAL) can only be set for the following devices:

- Resistor
- Capacitor
- Inductor
- BJT
- JFET
- MESFET
- MOSFET (level 1, 2, and 3)
- Diode

When using these parameters a maximum of one device temperature customization (T_ABS, T_REL_LOCAL, T_REL_GLOBAL) can coexist with the T_MEASURED parameter.

Note The Cadence® PSpice® temperature parameters have precedence over the TEMP instance and TNOM model parameters.

XSPICE Syntax Reference

XSPICE Code Model

XSPICE Code Model

XSPICE syntax for a terminal:

```
<%TerminalType> <(> nodename <, referencenodename> <)>

<%TerminalType> [ arraynodename1 <, arraynodename2 <,
...>> ]
```

XSPICE Terminal types:

Symbol	Description
%D	Digital terminal.
%V	Voltage terminal.
%I	Current terminal.
%G	Voltage in/current out terminal.

Symbol	Description
%H	Current in/voltage out terminal.
%VD	Differential voltage terminal.
%ID	Differential current terminal.
%GD	Differential voltage in/current out terminal.
%HD	Differential current in/voltage out terminal.
%VNAM	Named vsource “terminal”.

XSPICE code model instance line syntax:

```
Axxx xspiceterminal1 < xspiceterminal2 <...>> Model
<Instance_Parameters…>
```

XSPICE code-model model definition syntax:

```
.MODEL mymodelname CODEMODELNAME ( <Model_Parameters…> )
```

Please refer to XSPICE documentation at `http://users.ece.gatech.edu/~mrichard/Xspice` for further details.

Unusual Forms of Device Syntax

The syntax of the various devices listed above are given with the most common forms. However, there are several other variations which are also acceptable for compatibility reasons. In general, the following rules can be used:

- Brackets that are not part of a mathematical expression are optional.
- Commas between parameters are optional (except in mathematical functions)
- Equal signs on certain parameters are optional; they can be left out of places where they would normally go and added in after parameters names that don’t normally need them. The parameters that are optional are V, I, DC, AC, PWL, SIN, TABLE, and so on, in sources.
- Parameter names that are implied can also be stated explicitly (for example, “Resistance” on a resistor.)
- The order of explicitly stated parameters doesn’t matter.
- Except in mathematical expressions, whitespace doesn’t matter.

- Putting curly brackets, {}, around expressions is recommended but is not required in most cases. Inside expressions curly brackets, {}, can be used like parenthesis, ().

  In sources of the form similar to SIN and PWL, the brackets don't matter—these can be set with a line like SIN = 1 2 3 … in the same way that any other parameter, or the brackets, can be used to make logical pairs for PWL = (1,2),(3,4),(5,6).

Examples

Thus the following statements are acceptable input to the Multisim SPICE simulator:

```
R1 1 0 RESISTANCE=1k

V1 1 0 AC 2, 1 DC = 10

H1 1 2 POLY 1 V2 3.40k

I2 1 0 PWL =  0 0 1.3  2.0, 5.0,   5.0

E2 5 8 TABLE V(4)**2 0.0, 0.0, 0.5,0.5, 1.0,   0.0
```

Compatibility Modes

Some other forms of SPICE have slight differences which cannot be automatically detected by Multisim. In order for Multisim to correctly handle these cases it is necessary to add the following special command:

.SYNTAX mode

where mode can be PS for Cadence® PSpice®, XS for XSPICE, and MS for Multisim.

When this command is used, all other elements of the netlist that occur below it (including nested subckt definitions) will be interpreted with special compatibility mode until the end of the current subckt. The default compatibility mode is MULTISIM unless otherwise specified.

Default mode occurs when no .SYNTAX statements are present:

- The mathematical function log is the natural logarithm, except in 'B' arbitrary sources where it is the base 10 logarithm.

MS and XS mode are currently the same:

- The mathematical function log is the base 10 logarithm.
- The mathematical operator `**` is the standard mathematical operator ‘power’ and is exactly equivalent to the operator `^`.
- Mathematical expressions do not need to be surrounded by `{}`, however this is still recommended for clarity.

The PS mode has a few differences.

- The mathematical function log is the natural logarithm.
- The mathematical operator `**` is the same as the function PWR, which is equivalent to `abs(x)**y` in Multisim compatibility mode.
- The mathematical operator `^` is xor whereas in Multisim compatibility mode it is another way of writing `x**y`.
- The step function `STEP(x)` or `STP(x)` or `U(x)` is defined as `if(x>=0,1,0)`.
- Mathematical expressions other than numbers must be surrounded by {}.
- Equal signs are optional between all parameters and their values.
- When the character `*` is not inside {} it marks the start of a comment, even if it is in the middle of a line. Therefore in following example R1 will have a value of 1k in Cadence® PSpice® compatibility mode, but 2k in all other modes.

```
R1 1 2 1k*2
```

8

Simulation

This chapter explains the various types of simulation available in Multisim, the application for which each type is appropriate, how the types of simulation are used separately and together, and finally some of the underlying logic of Multisim simulation. It also details the circuit wizards that are available, and tools available for trouble-shooting simulation errors.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Introduction to Simulation

Simulation is a mathematical way of emulating the behavior of a circuit. With simulation, you can determine much of a circuit's performance without physically constructing the circuit or using actual test instruments. Although Multisim makes simulation intuitively easy-to-use, the technology underlying the speed and accuracy of the simulation, as well as its ease-of-use, is complex.

Multisim incorporates SPICE3F5 and XSPICE at the core of its simulation engine, with customized enhancements designed by National Instruments specifically for optimizing simulation performance with digital and mixed-mode simulation. Both SPICE3F5 and XSPICE are industry-accepted, public-domain standards. SPICE3F5 is the most recent edition of the SPICE (Simulation Program with Integrated Circuit Emphasis) core designed by the University of California at Berkeley. XSPICE is a set of unique enhancements made to SPICE, under contract to the US Air Force, which include event-driven mixed mode simulation, and an end-user extensible modelling subsystem. National Instruments has further enhanced these cores with certain non-SPICE-standard Cadence® PSpice® compatibility features to allow for using a wider range of off-the-shelf SPICE models.

Multisim's RF Design module simulates RF circuits using an optimized SPICE engine. There is no need to tell Multisim that your circuit is an RF circuit. RF simulation uses the SPICE simulation engine, but has been

optimized to accurately simulate circuits designed to operate at higher frequencies. This optimization uses parts specifically designed and modeled to simulate accurately at these higher frequencies.

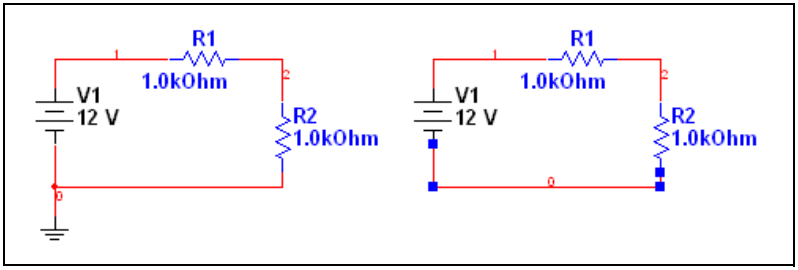
Using Multisim Simulation

You must use either a virtual instrument or run an analysis to display the simulation output. This output will include the combined results of all Multisim simulation engines.

When you use interactive simulation in Multisim (by clicking on the **Run/Resume Simulation** button), you see the simulation results instantly by viewing virtual instruments such as the oscilloscope. You can also view the effect of simulation on components like LED's and 7-segment digital displays.

As well as interactive simulation, you can run numerous analyses on your circuits. Results of analyses are displayed in the **Grapher** and can also be saved for later manipulation in the **Postprocessor**. Refer to the *Viewing the Analysis Results: Grapher* section of Chapter 10, *Analyses*, and the *Introduction to the Postprocessor* section of Chapter 11, *Postprocessor*, for more information.

Note All simulations require a reference net with respect to which all voltages are given. In SPICE, this is always net 0. Therefore, somewhere in the circuit a net named "0" must be defined. You may either name a net 0, or this will happen automatically if a ground component is wired into a circuit. This is illustrated in the figure below.



Interactive Components

Interactive components' values can be changed by pressing a specific key (set up in the component's **Value** tab) on your keyboard. You can change the value of interactive components during simulation and view the effect immediately. For example, changing a 100 kohm resistor to the next smaller resistor may alter the results more than desired, but with Multisim, you can use a variable resistor and reduce its value gradually while watching the changes to the simulation, until you reach the desired result.

Tip If you press a key, the value of all interactive devices on the schematic that are mapped to that key in their **Value** tabs will change. If you wish to change the value of a specific component only, hover the cursor over that component to display a control element. For example, if you hover the cursor over a potentiometer, a slider bar appears that you can slide to raise or lower the potentiometer's setting.

Interactive components include such devices as the potentiometer, variable capacitor, variable inductor, and switches. For details on specific interactive components, refer to the *Component Reference help file*.

Component Tolerances in Multisim

Multisim lets you include variances in your simulations that are introduced as a result of component tolerances. For example, a 1 kohm resistor with a 10% tolerance could vary plus or minus 100 ohms, which would in turn affect the results of its circuit's simulation.

Components that have user-settable tolerances are resistors, inductors, capacitors and some sources. It is not possible to set the tolerance for all resistors, all capacitors, etc. from one location. Tolerances must be set individually by following the procedures below.

Note **Worst Case** and **Monte Carlo** analyses will pick up the tolerances associated with the components. You can override these tolerances when setting up these analyses. Refer to the *Worst Case Analysis* and the *Monte Carlo Analysis* sections of Chapter 10, *Analyses*, for more information.

Note When loading circuits from old versions of Multisim, any tolerances from the old **Global Component Tolerances** dialog box are ignored and a message advising you of this appears in the **Results** tab of the **Spreadsheet View**.

Complete the following steps to set up the tolerance for a placed component:

1. Double-click on the component and select the **Value** tab.
2. Select or enter the desired value in the **Tolerance** field and click **OK**.

To set up tolerances in the **Spreadsheet View**, select the desired component in the **Components** tab of the **Spreadsheet View** and change the value in the **Tolerance** field.

Tip If you want to change the tolerance to the same value for a number of placed components, select them all in the **Spreadsheet View** using the <Ctrl> and <Shift> keys, and change the value in any **Tolerance** field.

Tolerances for resistors, inductors, or capacitors can also be set during part placement. Refer to the *Placing Resistors, Inductors or Capacitors* section of Chapter 2, *Schematic Capture - Basics*, for more information.

To use component tolerances during simulation, select **Simulate»Use Tolerances**. A checkmark appears beside the menu item.

Start/Stop/Pause Simulation

To simulate a circuit, click the **Run/resume simulation** button. Multisim begins to simulate the circuit's behavior. You can also select **Simulate»Run**.

Note When simulating a design that contains hierarchical blocks, subcircuits and/or multi-pages, the entire design is simulated, not just the current sheet. To simulate a hierarchical block in isolation, you must open that block as a new design by selecting **File»Open**. Subcircuits cannot be simulated by themselves.

During simulation, messages about the simulation results and any problems with the simulation are written to the simulation error log/audit trail. If you want to keep an eye on the progress of the simulation, you can display the error log/audit trail during simulation. Refer to the *Simulation Error Log/Audit Trail* section for more information.

To pause the simulation while it is running, select **Simulate»Pause**. To resume the simulation from the same point as when you paused, select **Simulate»Run**.

To stop a simulation, click the **Stop Simulation** button or select **Simulate»Stop**. If you restart the simulation after stopping it, it will restart

from the beginning (unlike **Pause**, which allows you to restart from the point where you paused).

Note Before running your simulation, you have the option to perform an electrical rules check. For details, refer to the *Electrical Rules Checking* section of Chapter 3, *Schematic Capture - Advanced Functions*.

Simulation Running Indicator

To indicate that a simulation is running, the **Simulation Running Indicator** appears in the status bar as shown in the figure below. This indicator flashes until you stop the simulation. This is especially useful when viewing an instrument that has reached a steady state, such as the **IV Analyzer**.

Simulation Speed

There are many parameters that affect simulation speed and convergence. These are accessible from the **Interactive Simulation Settings** dialog box. Some of the most important settings are shown on the first tab. Refer to the *Interactive Simulation Settings* section of Chapter 9, *Instruments*, for more information.

Note Convergence is the end-point of any algorithm that uses iteration or recursion to guide a series of data processing steps. An algorithm is usually said to have reached convergence when the difference between the computed and observed steps falls below a pre-defined threshold.

The most important simulation setting governing speed of simulation is TMax. TMax is the maximum time step that the simulator is allowed to take. In order to produce results, the simulator may take smaller time steps at its discretion, however it will never take a larger step than that specified by TMax. The smaller TMax is, the more accurate the simulation results will be. However it will take longer to reach any given simulation results.

In general, most simulations will run slower than real-time. However, if a lot of time resolution is not required, or if the circuit is primarily digital (in which case time steps will always be inserted where digital events occur and hence TMax may be set large) then TMax may be set to a higher value. If this results in the simulation running faster than real-time, it is artificially slowed down to real-time and your CPU is freed up for other tasks.

If “generate time steps automatically” is chosen, then TMax will be set in accordance with the highest frequency AC power source in the schematic, or to the smallest resolution instrument.

Circuit Consistency Check

When you simulate your circuit or perform an analysis, a circuit consistency check is performed to determine if the circuit obeys the simulation rules—for example, if a ground is present. Errors are written to the error log. This function speeds your simulation process, since it alerts you to items that may cause simulation errors and allows you to correct them before simulating.

Simulation from Netlist Without Schematic

You can also run simulations from a command line.

To open the command line interface, choose **Simulate»Xspice Command Line Interface**. The **XSpice Command Line** dialog box appears.

You can enter netlists and commands directly in this dialog box. The most important commands are: `SOURCE`, `PLOT`, `OP`, `SAVE`, `WRITE`, `TAN`, `SET` and `ANAC`.

This dialog works in the same manner as the **User Defined Analysis**. Refer to the *User Defined Analyses* section of Chapter 10, *Analyses*, for more information.

Multisim SPICE Simulation: Technical Detail

The following sections explain the basic technical methodology of circuit simulation in a SPICE-based simulator, including an outline of the stages of circuit simulation and an explanation of the methods of formulation and solution used in the circuit simulation. It is not necessary to fully understand this information to make use of Multisim’s simulation, but you may find it interesting.

To fully understand the information in this section, you should be acquainted with the theory of electronic circuit simulation and the mathematics involved.

Circuit Simulation Mechanism

After you create a circuit schematic and begin simulation, the solution of the circuit and generation of the data you see on instruments such as the oscilloscope is the role of the simulator. More specifically, the simulator is the part of Multisim that calculates a numerical solution to a mathematical representation of the circuit you created.

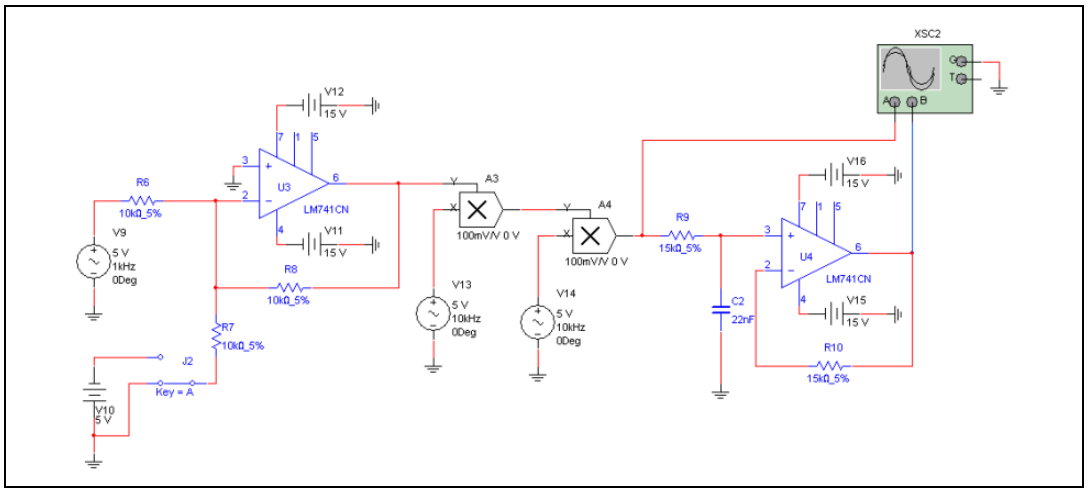
For this calculation to occur, each component in a circuit is represented by a mathematical model. Mathematical models link the schematic in the circuit window with the mathematical representation for simulation. The accuracy of the component models is one of the key items that determines the degree to which simulation results match real-world circuit performance.

The mathematical representation of a circuit is a set of simultaneous, nonlinear differential equations. The main task of the simulator is to solve these equations numerically. A SPICE-based simulator transforms the nonlinear differential equations into a set of linear algebraic equations. These equations are further linearized using the modified Newton-Raphson method. The resulting set of linear algebraic equations is efficiently solved using the sparse matrix processing LU factorization method.

Four Stages of Circuit Simulation

The simulator in Multisim has four main stages:

- **Input Stage**—Simulator reads information about your circuit (after you have built a schematic, assigned values and chosen an analysis). This is the process of netlist generation.
- **Setup Stage**—Simulator constructs and checks a set of data structures that contain a complete description of your circuit.
- **Analysis Stage**—The circuit analysis specified in the input stage is performed. This stage occupies most of the CPU execution time and is actually the core of circuit simulation. The analysis stage formulates and solves circuit equations for the specified analyses and provides all the data for direct output or postprocessing.
- **Output Stage**—You view the simulation results. You can view results on instruments such as the oscilloscope, on graphs that appear when you run an analysis, or in the log file/audit trail.



SPICE-based simulation works by first converting a schematic into a SPICE netlist.

The text below is a small part of the netlist that Multisim generates for the Amplitude Modulator circuit shown in the above figure. This netlist is the actual input to SPICE required to perform the simulation.

```
rR10 15 16 15000 vresR10
.model vresR10 r(  )
rR9 13 14 15000 vresR9
.model vresR9 r(  )
cC2 14 0 2.2e-008
xU4 14 15 16 LM741__OPAMP__1
```

Prior to the existence of schematic entry programs such as Multisim, designers were required to tediously create such SPICE netlists for themselves each time they wished to analyze a circuit. A modern schematic capture tool, in addition to providing a front-end for PCB layout or other downstream activities such as IC or FPGA design, does this time-consuming and error-prone task automatically.

From the netlist, SPICE generates matrices that it solves numerically to come up with voltages (AC and DC) at every node in the circuit. Current *branches* also appear in the matrices when required in order to solve the equations. In particular, current branches appear whenever voltage sources are used (SPICE trick: if you need to measure a current in SPICE, insert a 0V voltage source. It will not affect the circuit but will force SPICE to compute the current running through the 0V source).

For a transient analysis (which is also the basis for interactive simulation), the matrices are solved at every time step in the simulation. Because non-linear analog parts are present in the simulation, at each time step successive approximations are used to compute the final node voltage results. It is possible under certain circumstances that these results will not converge. When this happens, SPICE backs up to 1/8th the time step used previously and tries again.

Equation Formulation

In a circuit, each common point created by wires and connectors is called a node. The simulator calculates the voltage at each node. Each branch joining two nodes will have a separate current flowing through it.

To calculate a circuit solution, a circuit is represented internally as a system of equations, in the form:

$A*X = B$

where:

A = modified nodal admittance matrix with dimension $n \times n$.

X = vector of unknowns with dimension n

B = vector of constants, also with dimension n.

n = number of unknowns.

The system of equations is formulated using a general circuit analysis method called the Modified Nodal Approach (MNA).

The unknowns (*n)* include each node voltage (excluding ground), as well as the voltage source currents. B contains the voltage and current source constants, and the entries in the admittance matrix (A) are determined by Ohm's law and Kirchhoff's current and voltage laws.

The modified nodal admittance matrix is deemed sparse because it contains more zeros than non-zeros. Making use of a linked list, the solution of circuit equations can be performed by employing non-zero terms only. This method is called Sparse Matrix Technique. Generally, a sparse matrix approach requires less memory consumption and achieves faster simulation.

Equation Solution

Multisim solves circuit equations for linear and nonlinear circuits using a unified algorithm. The solution of a linear DC circuit is treated as a special case of general nonlinear DC circuits.

LU factorization is used to solve the system of sparse modified nodal matrix equations described previously (a set of simultaneous linear equations). This involves decomposing the matrix A into two triangular matrices (a lower triangular matrix, L, and an upper triangular matrix, U) and solving the two matrix equations using a forward substitution and a backward substitution.

Several efficient algorithms are used to avoid numerical difficulties due to the modified nodal formulation, to improve numerical calculation accuracy and to maximize the solution efficiency. These include:

- A partial pivot algorithm that reduces the round-off error incurred by the LU factorization method.
- A preordering algorithm that improves the matrix condition.
- A reordering algorithm that minimizes nonzero terms for the equation solution.

A nonlinear circuit is solved by transforming it into a linearized equivalent circuit at each iteration and iteratively solving the linear circuit using the above-described method. Nonlinear circuits are transformed into linear ones by linearizing all nonlinear components in the circuit using the modified Newton-Raphson method.

A general nonlinear dynamic circuit is solved by transforming the circuit into a discretized equivalent nonlinear circuit at each time point and solving it using the method for a nonlinear DC circuit described above. A dynamic circuit is transformed into a DC circuit by discretizing all dynamic components in the circuit using an appropriate numerical integration rule.

User Setting: Maximum Integration Order

You can change the maximum order for integration method using the MAXORD analysis option (refer to the *Custom Analysis Options Dialog Box* section of Chapter 10, *Analyses*, for information). Using a higher order (3 through 6) Gear method theoretically leads to more accurate results, but slows down the simulation. Be aware that the maximum order for integration method is the maximum order that could be used, but that the simulator selects the most appropriate order based on the circuit.

Due to the nature of the nonlinear components, each time point may involve solving the admittance matrix several times before converging to a solution. The point solution is reached when the difference between consecutive voltage values is less than the tolerance calculated internally in terms of the absolute and relative tolerances specified in the analysis options.

DC Convergence Assistance Algorithms

Multisim contains four continuation method algorithms to help find the solution of non-linear circuits during an Operating Point analysis. These algorithms are Static Diagonal Gmin Stepping, Static Source Stepping, Dynamic Diagonal Gmin Stepping, and Dynamic Source Stepping.

Continuation methods divide the process of finding the operating point into a series of manageable tasks. To start, the circuit is set to a state in which it is more easily solvable. Next, through a series of iterations, where the solution at one step becomes the entry point to the solution attempt at the next, it is gradually modified back to its original state. The specific changes made to the circuit during the process are what differentiate the four continuation methods.

Diagonal Gmin Stepping Algorithms

Both the Dynamic and Static algorithms change the diagonal elements of the modified nodal conductance matrix. The basic concept is to keep the matrix well-conditioned.

Static Diagonal Gmin Stepping starts by adding a large helper value to the conductance matrix diagonal and attempting a solution. The helper value is initially set to GMIN * $10^{GminSteps}$, where GMIN (Junction Minimum Conductance) and GMINSTEPS are obtained from the analysis options (refer to the *Custom Analysis Options Dialog Box* section of Chapter 10, *Analyses*, for more information on both of these options).

If a solution is found, the helper value is reduced by a factor of 10 and the circuit is solved again. This procedure is repeated with the goal of reducing the helper value to GMIN only. If that state is reached, the helper value is set to zero, bringing the circuit to its original state, and a final solution of the circuit is performed. The drawback of this algorithm is that, if at any step a solution cannot be found, the algorithm aborts.

The Dynamic Diagonal Gmin Stepping algorithm addresses this deficiency. Similar to the Static version, the Dynamic algorithm starts by adding a large helper value to the diagonal of the conductance matrix. A solution is attempted and if one is found, the helper value is decreased by the value of

the DYNAMICDCFACTOR option. This process is repeated in an attempt to reduce the helper value to GMIN. However, unlike the Static version, the Dynamic algorithm does not abort if a solution cannot be found at any one step. Instead, the helper value is decreased by a smaller factor. Only when this factor is reduced from 10 to approximately 1 does the algorithm abort. This process is analogous to the Transient Analysis algorithm, where the step is decreased when a solution cannot be found at a particular time point.

Source Stepping Algorithms

Both the Static and the Dynamic algorithms modify the DC value of independent voltage and current sources.

The Static Source Stepping algorithm starts by setting the DC value of all independent voltage and current sources to zero and attempting a solution. If a solution is found, the DC values are gradually increased to their nominal values in steps.

The step size is:

$$\frac{nominal\ \ value}{ITL6}$$

where ITL6 = number of source steps.

The drawback of this algorithm is that if at any one step a solution cannot be found, the algorithm aborts.

The Dynamic Source Stepping algorithm addresses this shortcoming. As in the Static version, the DC values of all independent voltage and current sources are set to zero and a solution is attempted. If a solution is found, the DC sources are then set to a fraction of their nominal values and a solution is again attempted. The values are gradually increased in steps until they are back to their nominal values.

However, unlike the static version, the Dynamic algorithm does not abort if a solution cannot be found at any one step. Instead, a smaller step size is used. The algorithm only aborts when the step size approaches zero. This process is analogous to the Transient Analysis algorithm, where the time step is cut when a solution may not be found at a particular time point.

Execution Order

If the DYNAMICDC option is set, the Dynamic Diagonal Gmin Stepping algorithm is attempted first—if it fails, the Dynamic Source Stepping algorithm is attempted. If that fails, the Operating Point analysis aborts.

If the DYNAMICDC option is not set, the Static Diagonal Gmin Stepping algorithm is attempted first—if it fails, the Static Source Stepping algorithm is attempted. If that fails, the Operating Point analysis aborts.

Note Neither the Static nor the Dynamic algorithms are attempted if the basic DC algorithm succeeds in finding the operating point.

Digital Simulation

Digital parts are modeled differently than analog parts. Digital parts are connected to the analog parts of the circuit using special XSPICE code models for A-to-D and D-to-A bridges. These models transform voltages into digital events and vice-versa. Thus a net in a schematic may be either analog or digital, but not both at once. When digital parts are connected to one another, the digital events propagate from one to the other with the appropriate simulated time delays. Time steps are automatically inserted into the simulation whenever digital events make their presence felt on the analog parts of the circuit. This event-driven simulation approach to digital simulation allows these types of simulation to run much more quickly than analog simulations (hint: set the simulation parameter Tmax, the maximum analog time step, quite high for digital simulations in order to speed things up dramatically).

When simulating circuits with digital components, you have the option of simulating for speed or for accuracy. The “Ideal” option simulates your circuit quickly by not taking into account variances in digital power and internal tolerances. The time to simulate digital components is faster but the signal is not as accurate.

The “Real” option simulates your circuit accurately, but slower than the “Ideal” option, by accounting for all variances. When using “Real” simulation settings, you are required to add digital power and digital ground to your circuit.

Complete the following steps to select a digital simulation option for the active design:

1. Choose **Simulate»Digital Simulation Settings** to display the **Digital Simulation Settings** dialog box.
2. Select one of:
   - **Ideal**—In this mode, if two digital pins are connected to one another, no additional circuitry is supplied. If a digital pin is attached to an analog node, a simplified pin driver circuit and A-to-D or D-to-A bridge is added into the netlist to smooth the edges of the abrupt digital transitions.
   - **Real**—In this mode, all digital pins are attached to D-to-A or A-to-D converters, and more complex pin drivers better representing actual digital pin drivers are inserted into the netlist.

Note "Real" mode provides for more accurate voltage results, but "Ideal" mode runs faster. This switch only affects circuits that contain digital parts.

3. Click **OK**.

Note You can change the default digital simulation settings from the **Preferences** dialog box. Refer to the *Preferences - Parts Tab* section of Chapter 1, *User Interface*, for more information.

RF Simulation

You simulate an RF circuit the same way you simulate a board/system-level circuit in Multisim. Refer to the *Using Multisim Simulation* section for more information. This is because Multisim's RF Design module simulates RF circuits using an optimized SPICE engine (as opposed to VHDL, etc.). There is no need to tell Multisim that your circuit is an RF circuit. RF simulation uses the SPICE simulation engine, but has been optimized to accurately simulate circuits designed to operate at higher frequencies, or at faster clock speeds (which generate RF characteristics). This optimization uses parts specifically designed and modeled to simulate accurately at these higher frequencies.

Refer to Chapter 14, *RF*, for information on RF simulation and RF Design Module.

MultiVHDL

VHDL (Very High Speed Integrated Circuit (VHSIC) Hardware Description Language) is designed to describe the behavior of complex digital devices. For this reason it is referred to as a "behavioral level" language; it can use behavioral level models (instead of transistor/gate level, like SPICE) to describe the behavior of these devices. Using VHDL avoids the unwieldy task of describing such devices at the gate level, greatly simplifying the design process.

MultiVHDL can be used in two ways:

- As part of the board/system design process, when components are modeled in VHDL instead of SPICE. Multisim automatically invokes the VHDL simulator as needed (this is called co-simulation). In this method, you do not need extensive VHDL knowledge, but can simply take advantage of the broader database of simulatable models for complex digital chips.
- As a VHDL source code editor/simulator, for writing and debugging VHDL source code.

For details, refer to the user documentation included with your legacy copy of MultiVHDL.

Circuit Wizards

Multisim's circuit wizards let you generate circuits that contain a schematic diagram, simulation models and a netlist. You simply enter the design parameters in a wizard's dialog box and click a button to build the circuit. Once you have built the circuit, it can be simulated in the usual manner. The wizards that are available are:

- *555 Timer Wizard*
- *Filter Wizard*
- *Common Emitter BJT Amplifier Wizard*
- *Op-amp Wizard*

555 Timer Wizard

Use the **555 Timer Wizard** to build astable and monostable oscillator circuits that use the 555 timer.

- **Astable Operation**—Produces a free-running oscillator that does not need any input signal.
- **Monostable Operation**—Produces a single output pulse in response to an input trigger pulse. When an input signal is applied, each input pulse will produce one output pulse.

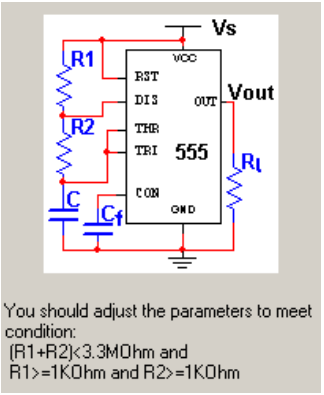
Complete the following steps to build an astable oscillator:

1. Select **Tools»Circuit Wizards»555 Timer Wizard**. The **555 Timer Wizard** dialog box appears.
2. Select **Astable Operation** from the **Type** drop-down list. Default settings are initially displayed.
3. Enter values based on the following:
   - **Vs**—Enter desired source voltage.
   - **Frequency**—Enter the frequency at which you wish the circuit to oscillate, to a maximum of 1 MHz.
   - **Duty**—Enter the duty cycle for the circuit. For example, 60% indicates that the circuit will be “on” for 60% of each cycle. Value must be greater than or equal to 50% and less than 100%.
   - **C**—This is the value of capacitor C and is initially set to 10 nF.
   - **Cf**—This is the value of capacitor Cf and is normally fixed to be 10 nF.
   - **Rl**—Enter desired load resistance.
4. As values are entered, R1 and R2 are calculated automatically and changed based on the following:

   R2 = (1-d)/(0.693*f*C)

   If d = 0.5, R1 = R2/80, otherwise, R1 = (2*d-1)*R2/(1-d)

   where d is the duty cycle, f is the oscillation frequency and C is the value of capacitor C.
5. If (R1+R2)>3.3 Mohm or R1<1 kohm or R2<1 kohm, a warning message displays in the **555 Timer Wizard** dialog box, as shown below the schematic preview in the example below.



If this happens, change the value of capacitor C and other parameters until the error message no longer displays.

6. Check the values of R1 and R2. If they are unavailable or unsatisfactory, change the value of capacitor C.
7. Repeat the above two steps until R1 and R2 are satisfactory and there is no error message in the **555 Timer Wizard** dialog box.
8. Click on the **Build Circuit** button. The circuit, with the calculated component values is placed on your workspace.

Complete the following steps to build a monostable (one-shot) oscillator:

1. Select **Tools»Circuit Wizards»555 Timer Wizard**. The **555 Timer Wizard** dialog box appears.
2. Select **Monostable Operation** from the **Type** drop-down list.
3. Enter values based on the following:
   - **Vs**—Enter desired source voltage.
   - **Vini**—Set equal to Vs.
   - **Vpulse**—Enter desired voltage of input pulse. This should be less than Vs/3.
   - **Frequency**—Enter the frequency of the input voltage.
   - **Input Pulse Width**—Enter desired input pulse width. Must be less than Output Pulse Width/5. If it is not, the value in the **Output Pulse Width** field is changed by the system.
   - **Output Pulse Width**—Enter desired output pulse width.
   - **C**—This is the value of capacitor C and is initially set to 1 uF.

- **Cf**—This is the value of capacitor Cf and is normally fixed to be 1 nF.
- **Rl**—Enter desired load resistance.

4. As values are entered, R is calculated automatically and changed based on the following:

   R = OW/(1.1*C)

   where OW is the output pulse width and C is the value of capacitor C.
5. Check the value of R. If it is unavailable or unsatisfactory, change the value of capacitor C or other parameters until a satisfactory value is reached.
6. Click the **Build Circuit** button.
7. The circuit, with the calculated component values is placed on your workspace.

Filter Wizard

The Multisim **Filter Wizard** lets you design numerous types of filters by entering the desired specifications into its fields. The designed circuit can then be verified by SPICE simulation.

Complete the following steps to design a filter using the **Filter Wizard**:

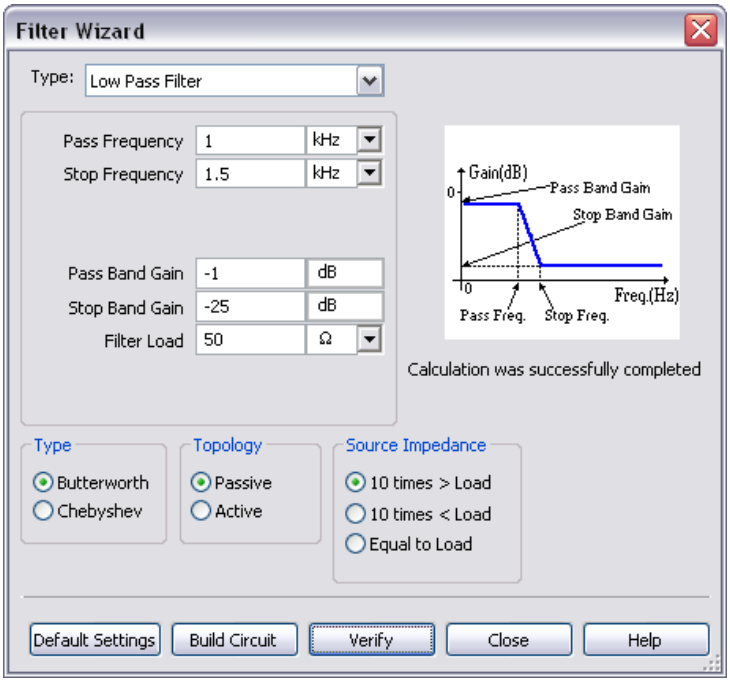
1. Select **Tools»Circuit Wizards»Filter Wizard**.
2. Select the desired filter type from the **Type** drop-down list. Choices are low pass, high pass, band pass or band reject. The parameters that are available in the box below the **Type** drop-down and the diagram on the right of the dialog box change based on the selected filter type and the selections made in the following steps.
3. Select either **Butterworth** or **Chebyshev** in the **Type** box. If you select **Chebyshev**, the **Pass Band Ripple** box displays. If you select **Butterworth**, it does not.
4. Select either **Passive** or **Active** in the **Topology** box. If you select **Passive**, the **Source Impedance** box displays. If you select **Active**, it is does not.
5. Select the desired source impedance in the **Source Impedance** box (for **Passive** filters only).
6. Select the desired ripple in the **Pass Band Ripple** box (for **Chebyshev** type filters only).

Note The **Pass Band Ripple** box does not appear if **Equal to Load** is selected in the **Source Impedance** box.

7. Enter desired filter parameters in the box below the **Type** drop-down list.

Note The available parameters change depending on the selections made in the above steps.

8. Click **Verify**. If there are any problems with your design, a message displays below the diagram in the **Filter Wizard** dialog box. Adjust your parameters and click **Verify** again. When your design is error-free, a "calculation successful" message displays and the **Build Circuit** button becomes active:



9. Click **Build Circuit**. The **Filter Wizard** dialog box closes and a "ghost" image of the circuit is placed on your cursor.
10. Move the cursor to the desired location and click the mouse to place the circuit.

Common Emitter BJT Amplifier Wizard

The Multisim **Common Emitter BJT Amplifier Wizard** lets you design common emitter amplifier circuits by entering the desired specifications into its fields. The designed circuit can then be verified by SPICE simulation directly.

Complete the following steps to design a common emitter amplifier using the **Common Emitter BJT Amplifier Wizard**:

1. Select **Tools»Circuit Wizards»CE BJT Amplifier Wizard**. The **BJT Common Emiter Amplifier Wizard** dialog box displays.
2. Enter the desired parameters in the **BJT Selection**, **Amplifier Specification**, **Quiescent Point Specification**, and **Load Resistance and Power Supply** boxes.
3. Click **Verify**. The values in the **Amplifier Characteristics** box change based on the following:

```
if in type "Ic"
{
ic_temp = Ic;
rb_temp = 0.1 * Hfe * (Vcc / ic_temp - 1.5 * Rl);
}
else if in type "Vce"
{
ic_temp = Vce / (Rl / 2);
rb_temp = 0.1 * Hfe * (Vcc / ic_temp - 1.5 * Rl);
}
else if in type "Vpswing"
{
ic_temp = (Vpswing + 0.2 + Vpin) / (Rl / 2);
rb_temp = 0.1 * Hfe*(Vcc / ic_temp - 1.5 * Rl);
}

rpi_temp = Hfe * 26.0e-3 / ic_temp;
rin_temp = rpi_temp * rb_temp / ( rpi_temp + rb_temp
);
avp_temp = ic_temp * Rl / ( 26e-3 * 2 );

av = (avp_temp * rin_temp / ( rin_temp + Rs);
ai = rb_temp * Rl / ((rb_temp / Hfe + rpi_temp / Hfe)
* ( 2 * Rl ));
avmax = Vcc/26e-3;
```

4. If there are any problems with your design, a message displays. Adjust your parameters and click **Verify** again.
5. Click **Build Circuit**. The **BJT Common Emitter Amplifier Wizard** dialog box closes and a "ghost" image of the circuit is placed on your cursor.
6. Move the cursor to the desired location and click the mouse to place the circuit.

Op-amp Wizard

The Multisim **Op-amp Wizard** lets you design the following op-amp circuits by entering the desired specifications into its fields:

- Inverting Amplifier
- Non-inverting Amplifier
- Difference Amplifier
- Inverted Summing Amplifier
- Non-inverted Summing Amplifier
- Scaling Adder.

The designed circuit can then be verified by SPICE simulation directly.

Tip When working in the **Op-amp Wizard**, you can return to the default settings at any time by clicking the **Default Settings** button.

Complete the following steps to build a circuit with the **Op-amp Wizard**:

1. Select **Tools»Circuit Wizards»Op-amp Wizard** to display the **Op-amp Wizard** dialog box.
2. Select the type of circuit you wish to build from the **Type** drop-down list. The contents of the dialog box and the diagram in the preview area change to reflect the selection.
3. If you do not wish to include a source in the circuit, de-select the **Add Source** checkbox. For inverted summing amplifiers, non-inverted summing amplifiers, or scaling adders, you must also enter the **Number of Inputs**.
4. In the **Input Signal Parameters** box, set the desired **Input Voltage** and **Input Frequency** values.

Note The number of **Input Voltage** and **Input Frequency** fields differ depending on the selection you made in the **Type** drop-down list, and also on the value you entered in the **Number of Inputs** field. These fields are not active if you de-selected **Add Source**.

5. Enter the desired values in the **Amplifier Parameters** box.

Note The available **Amplifier Parameters** change according to the selection you make in the **Type** drop-down list and the value you entered in the **Number of Inputs** field.

6. Click **Verify**. If a warning message displays, you may wish to adjust your parameters and click **Verify** again.
7. Click **Build Circuit**. The **Op-amp Wizard** dialog box closes and a "ghost" image of the circuit is placed on your cursor.
8. Move the cursor to the desired location and click the mouse to place the circuit.

Netlist and Simulation Errors

The **Simulation** tab of the **Spreadsheet View** is where errors and warnings from SPICE netlist checks, as well as simulation errors, appear.

Note With the exception of the **Simulation** tab and the **Results** tab, the **Spreadsheet View** is not available in all editions of Multisim. Refer to the release notes for information on what features are included with your software. Refer to the *Spreadsheet View* section for more information about all tabs.

Netlist Errors and Warnings

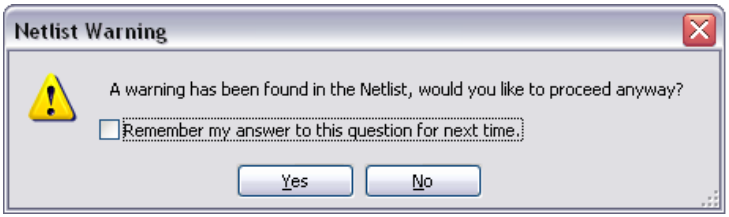
When you start a simulation or open an analysis dialog box, Multisim runs a SPICE netlist check and reports any netlist warnings or errors in the **Simulation** tab of the **Spreadsheet View**. If the **Simulation** tab is not already open, it opens at this point. The tab does not open if the netlist check is error and warning-free.

Netlist Warning Messages

The following is an example of a netlist warning message:

```
SPICE Netlist Warning in schematic RefDes 'u4', element
'c':  Multiple values have been assigned to non-array
parameter 'capacitance'; only the last one will be used
(check for spaces or tabs in your expression)
```

By default, when a warning displays in the **Simulation** tab, the **Netlist Warning** dialog box also appears, as shown below:


Netlist Warning
A warning has been found in the Netlist, would you like to proceed anyway?
Remember my answer to this question for next time.
Yes
No

Complete the following steps when the **Netlist Warning** dialog appears:

1. Optionally, enable the **Remember my answer to this question for next time** checkbox.
2. Click **Yes** to continue the simulation or analysis, or **No** to cancel.

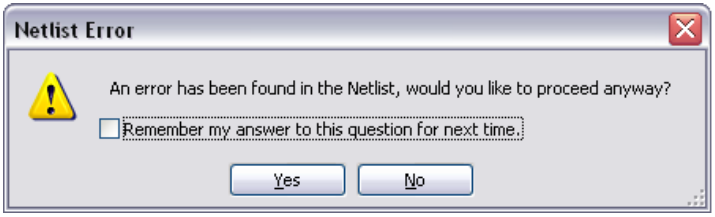
Note This is the default behavior for netlist warnings. You can change this to cancel the simulation or analysis, or to continue with the simulation or analysis without showing the **Netlist Warning** dialog box. Refer to the *Preferences - General Tab* section for more information.

Netlist Error Messages

The following is an example of a netlist error message:

```
SPICE Netlist Error in schematic RefDes 'u4', element
'g3':  Not enough nodes found
```

By default, when an error displays in the **Simulation** tab, the **Netlist Error** dialog box also appears, as shown below:


Netlist Error
An error has been found in the Netlist, would you like to proceed anyway?
Remember my answer to this question for next time.
Yes
No

Complete the following steps when the **Netlist Error** dialog appears:

1. Optionally, enable the **Remember my answer to this question for next time** checkbox.
2. Click **Yes** to continue the simulation or analysis, or **No** to cancel.

Note This is the default behavior for netlist errors. You can change this to cancel the simulation or analysis, or to continue with the simulation or analysis without showing the **Netlist Error** dialog box. Refer to the *Preferences - General Tab* section for more information.

Simulation Errors

The **Simulation** tab also displays simulation errors, as in the example below:

```
Error Message From Simulation: doAnalyses: Timestep too
small
```

Open the **Simulation Error Log/Audit Trail** dialog box to view the error in more detail. Refer to the *Simulation Error Log/Audit Trail* section for more information.

Note Simulation warnings and messages do not appear in the **Simulation** tab—you must open the **Simulation Error Log/Audit Trail** dialog box to view these.

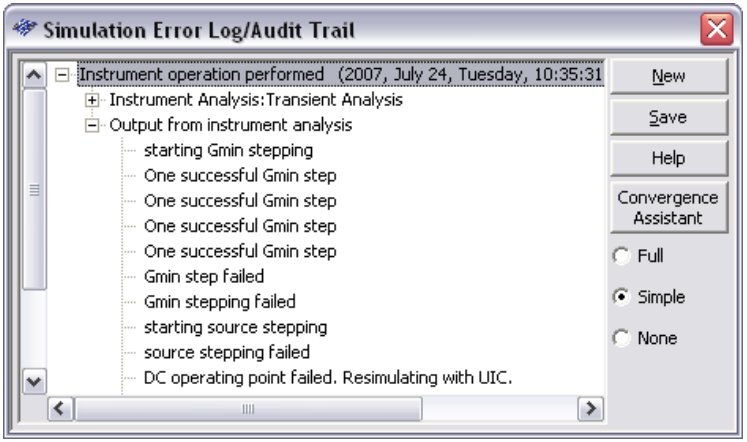
Simulation Error Log/Audit Trail

You can view the results of a simulation in the **Simulation Error Log/Audit Trail** dialog box.

If you turn on the analysis option ACCT, the **Simulation Error Log/Audit Trail** dialog box also includes errors or warning messages generated during simulation, and a chart of simulation statistics. Refer to the *Custom Analysis Options Dialog Box* section of Chapter 10, *Analyses*, for more information about analysis options.

To display the **Simulation Error Log/Audit Trail** dialog box, select **Simulate»Simulation Error Log/Audit Trail**.

The **Simulation Error Log/Audit Trail** dialog box is useful for diagnosing errors that occur during analysis or interactive simulation. Each analysis you perform, either individually or in a batch, during this Multisim session, as well as each interactive simulation, is stored in the audit trail. The file is cleared when you exit Multisim. A sample display is shown in the figure below.



To expand or contract the contents of the display to show or hide details, click on the "**+**" or "**–**" in the tree that appears in the **Simulation Error Log/Audit Trail** dialog box.

Complete the following steps to choose the level of errors to display:

1. Click one of the following buttons in the **Simulation Error Log/Audit Trail** dialog box:
   - **Full**—Displays all the errors.
   - **Simple**—Displays only the simple errors.
   - **None**—Display none of the errors.

To save the results of the audit trail in a separate file, click **Save** and choose a file name and location.

To clear the contents of the **Simulation Error Log/Audit Trail** dialog box, click **New**.

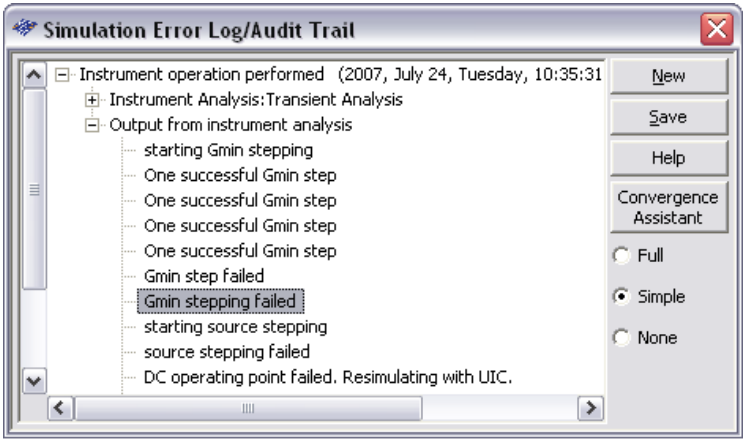
To display simulation error help, highlight the desired error in the audit trail and click **Help**. Refer to the *Simulation Error Help* section for more information.

Simulation Error Help

Simulation error help provides trouble-shooting information for errors that you may encounter during simulation.

Complete the following steps to show simulation error help for an error:

1. Select the desired error in the **Simulation Error Log/Audit Trail** dialog box as in the example shown in the figure below.



2. Click **Help**. A help topic with information for the selected error displays. Refer to the *Multisim help file* for details.

Convergence Assistant

Convergence is the end-point of any algorithm that uses iteration or recursion to guide a series of data processing steps. An algorithm is usually said to have reached convergence when the difference between the computed and observed steps falls below a pre-defined threshold.

When a convergence error occurs during interactive simulation in Multisim, a message appears asking if you would like to run the **Convergence Assistant** to try to automatically resolve the problem.

Complete the following steps to run the **Convergence Assistant**:

1. Click **Yes** when the above-mentioned message appears. The **Convergence Assistant** dialog box appears, and begins to attempt solutions to resolve the convergence issue.

2. Once the simulation error has been fixed, the **Convergence Assistant** rolls back the changes one-by-one, until error-free simulation occurs with the least possible amount of changes to the circuit's configuration. A summary report appears in the **Convergence Assistant** dialog box detailing the changes that were made.

You are also advised if the **Convergence Assistant** was unable to fix the error.

If you decline to run the **Convergence Assistant** when a convergence error occurs, an error message displays in the **Simulation** tab of the **Spreadsheet View**. Refer to the *Netlist and Simulation Errors* section for more information.

Complete the following steps to run the assistant from the **Simulation Error Log/Audit Trail** dialog box:

1. Click **Convergence Assistant** in the **Simulation Error Log/Audit Trail** dialog box.

Note The **Convergence Assistant** button only appears if a convergence error has occured.

2. Click **Start** in the **Convergence Assistant** dialog box.

If the **Convergence Assistant** is unable to fix the error:

1. Select the desired error in the **Simulation Error Log/Audit Trail** and click **Help**. Simulation error help appears, providing trouble-shooting suggestions for the selected error. Refer to the *Simulation Error Help* section for more information.

To save the contents of the **Convergence Assistant** log, click **Save**.

To print the contents of the **Convergence Assistant** log, click **Print**.

If you do not wish to keep the changes, click **Cancel**.

To keep the changes, click **OK**.

Saving/Loading Simulation Profiles

You can save simulation profiles that contain specific settings for the analyses you have used. These profiles can be used in other circuits instead of re-entering the settings for the various analyses.

Note Refer to the *Introduction to Multisim Analyses* section of Chapter 10, *Analyses*, for information about analyses.

Saving a Simulation Profile

Complete the following steps to save the simulation settings from the current circuit:

1. Select **Simulate»Save Simulation Settings**.
2. Navigate to the desired location, enter a filename for the profile, and click **Save**. The **Save Simulation Profile** dialog box displays. The filename that you just entered appears in the **Name** field.
3. Optionally, enter a **Description** of the profile.
4. Enable **Interactive Simulation Settings** if you wish to save custom settings that you entered in the **Interactive Simulation Settings** dialog box. Refer to the *Interactive Simulation Settings* section of Chapter 9, *Instruments*, for more information.
5. In the **Analyses** area, select the analyses that you wish to include in this profile.
6. Click **OK** to finish.

Loading a Simulation Profile

Complete the following steps to load an existing simulation profile:

1. Select **Simulate»Load Simulation Settings**, navigate to the desired profile and click **Open**. The **Load Simulation Profile** dialog displays.
2. Enable **Interactive Simulation Settings** if you wish to load the custom interactive simulation settings that you saved with the profile.
3. Select the **Analyses** whose saved profiles you would like to load and click **OK**.

9

Instruments

This chapter explains how to use the numerous virtual instruments provided as part of Multisim. It explains both the general procedures for attaching and configuring the instruments, and the specific steps for using each instrument. This chapter also explains how to create your own custom instruments using the National Instruments LabVIEW graphical development environment.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Introduction to the Multisim Instruments

Multisim provides a number of virtual instruments that you use to measure the behavior of your circuits. These instruments are set, used and read just like their real-world equivalents. They look just like the instruments you've seen and used in a lab. Using virtual instruments is the easiest way to examine your circuit's behavior and show the results of a simulation. In addition to the standard instruments that come with Multisim, you can create your own custom instruments using LabVIEW, a graphical development environment for creating flexible and scalable test, measurement, and control applications.

Virtual instruments have two views: the instrument icon you attach to your circuit, and the instrument face, where you set the instrument's controls. You can show or hide the face by double-clicking on the instrument's icon. The instrument faces are always drawn on top of the main workspace so that they are not hidden. You can place the instrument faces wherever you wish on your desktop. When you save your circuit, the instrument face locations and hide/show status are stored with the circuit. As well, any data contained in the instruments is saved, up to some maximum size (refer to the *Saving Simulation Data with Instruments* section for more information).

The instrument's icon indicates how the instrument is connected into the circuit. Once the instrument is connected to the circuit, a black dot appears inside the terminal input/output indicators on the instrument face.

Saving Simulation Data with Instruments

If **Save simulation data with instruments** in the **Save** tab of the **Preferences** dialog box is checked, the data displayed on the instrument faces will be saved in the circuit file (in addition to instrument settings and visibility state that are always saved). Because instruments such as the oscilloscopes may contain a lot of data, file sizes may become very large (despite the compression algorithms that are used during save). For this reason, you may set a maximum size threshold. If upon saving the maximum threshold is exceeded, you will be asked whether you wish to save the instrument data, save a subset of the instrument data, or save no instrument data.

Note Whether Multisim saves the data for a LabVIEW instrument depends on the individual instrument.

Adding an Instrument to a Circuit

Instruments are placed on the schematic from the **Instruments** toolbar. Refer to the *Instruments Toolbar* section of Chapter 1, *User Interface*, for more information.

This toolbar is displayed on the workspace by default. If it is not displayed, select **View»Toolbars»Instruments**. You may also right-click in any of the blank toolbar areas and select **Instruments** from the pop-up that appears.

(LabVIEW instruments appear in a sub-menu at the end of the **Instruments** toolbar.)

Complete the following steps to place an instrument in a circuit:

1. From the **Instruments** toolbar, click the button of the instrument you want to place.

   To add a LabVIEW instrument, click the arrow in the **LabVIEW Instruments** button and select the instrument you want to place from the sub-menu that appears, or use the **LabVIEW Instruments** toolbar.

   Refer to the *LabVIEW Instruments* section for more information on these instruments.

2. Move the cursor to the location on the circuit window where you want to place the instrument and click. The instrument is placed with the connections landing on the grid.

   The instrument icon and its reference designator (RefDes) appear. The RefDes identifies the type of instrument and its instance. For example,

the first multimeter you place on a circuit is called “XMM1”, the second is “XMM2”, and so on. This numbering is unique within each circuit—if you create a second circuit, the first multimeter placed in it is “XMM1”, and so on.

Note Some editions of Multisim do not support all instruments or multiple instances of a specific instrument.

3. To wire the instrument into the circuit, click on a terminal on the instrument’s icon and drag a wire to the desired location in the circuit (a pin, wire, or junction). The rules for component wiring also apply to instruments. Refer to *Wiring Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Note The voltmeter and the ammeter are found in the **Indicators** component group. They are not accessed by the method described above. Refer to the *Voltmeter* and *Ammeter* sections for information about these instruments, including how to connect them.

Using the Instrument

Complete the following steps to use the instrument:

1. To view and modify an instrument’s controls, double-click its icon. The instrument face appears. Make any necessary changes to the control settings, just as you would on their real-world equivalents. The control settings are different for each instrument, so if you are unfamiliar with them or need instruction, refer to the section on the specific instrument.

   It is critical that the control settings be appropriate for your circuit. If the settings are incorrect, this may cause the simulation results to appear incorrect or difficult to read.

Note Not all areas of the open instrument are modifiable. A hand appears when your cursor is on a control that can be modified.

2. To simulate the circuit, click the **Run/resume simulation** button on the **Simulation** toolbar. Multisim begins to simulate the circuit’s behavior and the signals, as measured at the points to which you have connected the instrument, are displayed.

   While the circuit is activated, you can adjust instrument settings. However, you cannot change the circuit by changing values (except for variable components), or perform any schematic capture functions

such as rotate or component replacement as these would invalidate the netlist.

You can also run, stop, pause, or resume using commands from the **Simulate** menu.

To stop the simulation, click the **Stop simulation** button on the **Simulation** toolbar. The simulation ends, with the final results shown on the instrument face and in the audit trail.

Working with Multiple Instruments

A single circuit can have multiple instruments attached to it, including (for some editions) multiple instances of the same instrument. In addition, each circuit window can have its own set of instruments. Setting up many different instruments or multiple instances of one instrument is done in exactly the same way as setting up one instrument.

Instruments that sample for a period of time cause a transient analysis to be run. If you use multiples of such instruments, only one transient analysis is run. The settings of this analysis are derived from considering all the concurrent instruments and choosing settings that will satisfy each. For example, if you have two oscilloscopes with two different time-bases (resolutions), Multisim uses the time-base of the oscilloscope with the smallest time-base (highest resolution). As a result, both instruments will sample at a higher resolution than they would individually.

The results from each instrument are recorded separately in the error log/audit trail.

Saving Instrument Data

You can set up your circuit so that data displayed on instruments are saved when the circuit is saved. Refer to the *Preferences - Save Tab* section of Chapter 1, *User Interface*, for more information.

Note Whether Multisim saves the data for a LabVIEW instrument depends on the individual instrument.

Printing Instruments

Multisim lets you print out the faces of selected instruments in your circuit. Any simulation data for the circuit appears on the printout.

Note Multisim does not print the faces of LabVIEW instruments.

Complete the following steps to produce a print-out of the instrument faces:

1. Have the desired circuit open in your workspace.
2. Select **File»PrintOptions»Print Instruments**. The **Print Instruments** dialog box appears. The instruments in your circuit are listed in the **Select Instruments** area.
3. Click on the checkbox beside any instrument to select/deselect it, then click **Print** to print the selected instruments.
4. A standard print dialog appears. Choose the desired print options and click **OK**.

Interactive Simulation Settings

Multisim lets you set default settings for instruments that are based on a transient analysis (such as the oscilloscope, spectrum analyzer and logic analyzer).

Complete the following steps to set the default instrument settings:

1. Choose **Simulate»Interactive Simulation Settings**. The **Interactive Simulation Settings** dialog box appears displaying the most commonly used functions in the **Defaults for Transient Analysis Instruments** tab.
2. Edit the following as required:
   - **Initial Conditions** drop-down list—Select one of: **Zero**, **User-Defined**, **Calculate DC Operating Point**, or **Automatically Determine Initial Conditions** (Multisim tries to start the simulation using the DC operating point as the initial condition. If the simulation fails, it uses user-defined initial conditions, which by default are set to 0).
   - **Start time (TSTART)** field—Start time of transient analysis must be greater than or equal to 0 and less than end time.
   - **End time (TSTOP)** field—End time of transient analysis must be greater than start time.

- **Set maximum timestep (TMAX)** checkbox—Enable to enter the maximum time step the simulation can handle.
- **Maximum timestep (TMAX)** radio button—Enable to manually set time steps.
- **Generate timesteps automatically** radio button—Enable to generate time steps automatically.
- **Set initial time step** checkbox—Enable to set a time interval for simulation output and graphing.
- **Estimate maximum time step tased on net list (TMAX)** checkbox—This becomes enabled when the **Maximum timestep (TMAX)** radio button is selected. Enable as desired.

3. Optionally, click the **Output** tab and disable the **Show all device parameters at end of simulation in the audit trail** checkbox. Disable this checkbox if the circuit takes a long time to exit simulation.
4. Click **OK**. These settings will be in effect the next time you run a simulation.

You can control many aspects of the simulation, such as resetting error tolerances, selecting simulation techniques, and viewing the results. The options you choose will determine the efficiency of the simulation. You set these options through the **Analysis Options** tab. Refer to the *Custom Analysis Options Dialog Box* section of Chapter 10, *Analyses*, for information about the analysis options and their default values.

Troubleshooting Simulation Errors

The default values in the **Interactive Simulation Settings** dialog box are set so the majority of circuits will simulate correctly. However in some cases, it will be necessary to adjust these values.

When running simulation, Multisim may encounter time-step difficulties depending on the circuit configuration. This may cause inaccurate simulation or simulation errors.

Complete the following steps to adjust the **Interactive Simulation Settings** values:

1. Start Multisim and load the circuit file exhibiting the problem.
2. Select **Simulate»Interactive Simulation Settings**.
3. Select the **Defaults for Transient Analysis Instruments** tab, and set the following:
   - Select **Set to zero** from the **Initial conditions** pull-down menu.

- Select **Maximum time step (TMAX)** and change the value to 1e-3 sec.

4. Click **OK** and run simulation.

If the problem persists, follow the additional steps below:

1. Select **Simulate»Interactive Simulation Settings**.
2. Select the **Analysis Options** tab, enable **Use Custom Settings** and click **Customize** to display the **Custom Analysis Options** dialog box.
3. In the **Global** tab, set the following:
   - Enable the **reltol** parameter and set the value to 0.01 (or for better accuracy try 0.0001).
   - Enable the **rshunt** parameter and set the value to 1e+8 (do this only if you are experiencing simulation error messages).
4. Click **OK** twice and run simulation.

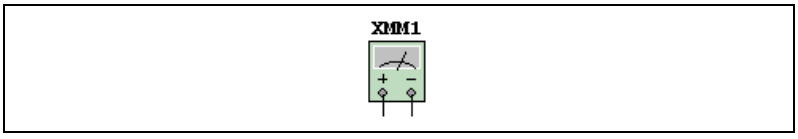
If the problem continues to persist, follow the additional steps below:

1. Select **Simulate»Interactive Simulation Settings**.
2. Select the **Analysis Options** tab, enable **Use Custom Settings** and click **Customize** to display the **Custom Analysis Options** dialog box.
3. In the **Transient** tab, set the following:
   - Enable the **METHOD** parameter and set it to **gear** from the pull-down menu.
4. Click **OK** twice and run simulation.

Multimeter

Use the multimeter to measure AC or DC voltage or current, and resistance or decibel loss between two nodes in a circuit. The multimeter is auto-ranging, so a measurement range does not need to be specified. Its internal resistance and current are preset to near-ideal values, which can be changed. Refer to the *Internal Settings - Multimeter Settings Dialog Box* section for more information.

To place the instrument, click on the **Multimeter** button in the **Instruments** toolbar and click to place its icon on the workspace. Use the icon, shown in the figure below, to wire the **Multimeter** to the circuit.



Double-click on the icon to open the instrument face, shown in the figure below. This is where you enter settings and view measurements.

Note You should read the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Multimeter Settings

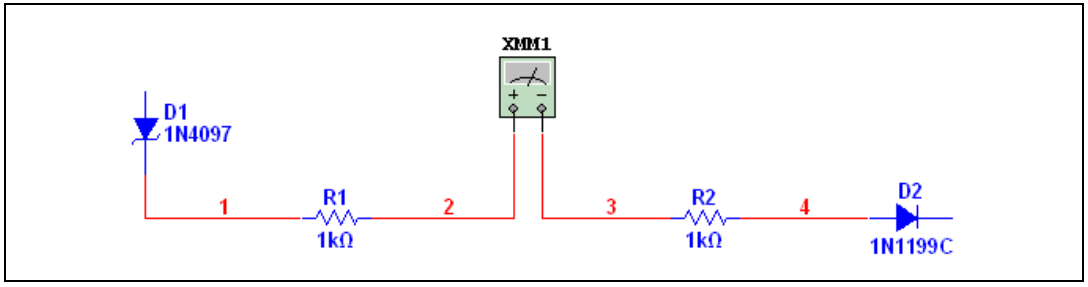
This section provides details on how to set up your multimeter.

Measurement Options

Complete the following steps to select the type of measurement to be taken:

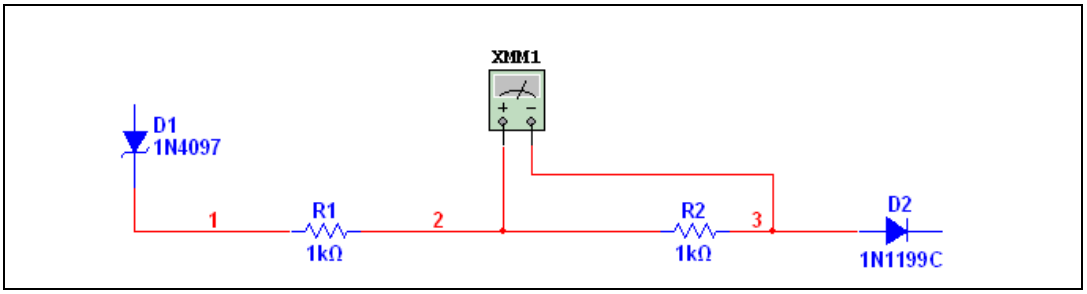
1. Click one of the following buttons:

Ammeter—Measures current flowing through the circuit in a branch between two nodes. Insert the multimeter in series with the load to measure current flow, as shown below.



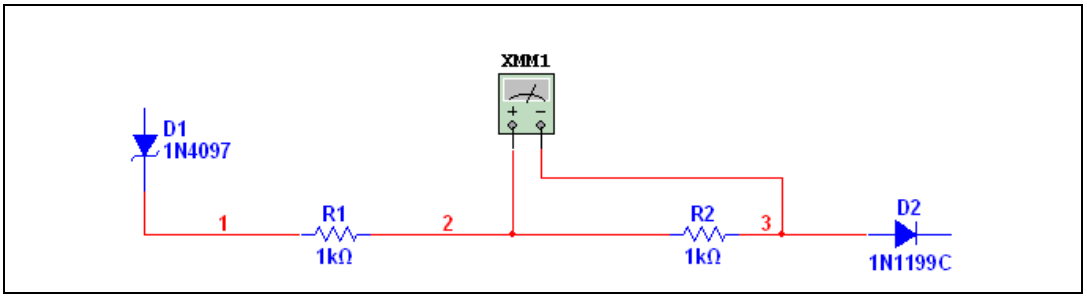
To measure current at another node in the circuit, connect another multimeter in series at that load and activate the circuit again. When used as an ammeter, the multimeter's internal resistance is very low (1 nohm). To change the resistance, click **Set**. Refer to the *Internal Settings - Multimeter Settings Dialog Box* section for more information.

Voltmeter—Measures voltage between two nodes. Select **V** and attach the voltmeter's probes in parallel with the load, as shown below.



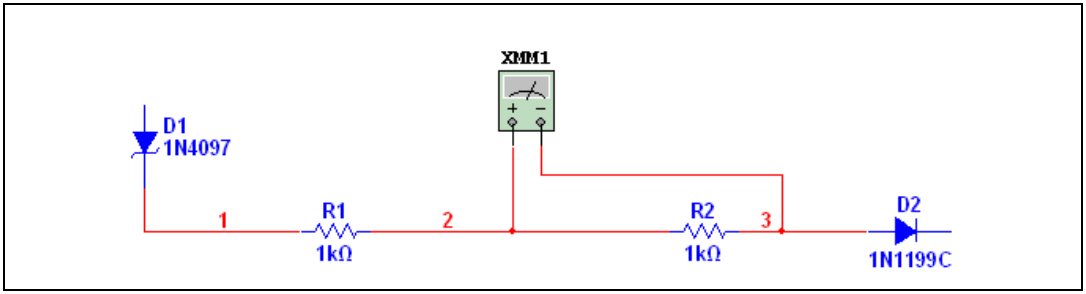
- When used as a voltmeter, the multimeter has a high internal resistance of 1 Gohm, which can be changed by clicking **Set**. Refer to the *Internal Settings - Multimeter Settings Dialog Box* section for more information.

Ohmmeter—This option measures resistance between two nodes. The nodes and everything that lies between them are referred to as the "component network". To measure the resistance, select this option and attach the multimeter's probes in parallel with the component network, as shown below.



- To get an accurate measurement, make sure that:
  - there is no source in the component network
  - the component or component network is grounded
  - there is nothing else in parallel with the component or component network.
- The ohmmeter generates a 10 nA current, which can be changed by clicking **Set**. Refer to the *Internal Settings - Multimeter Settings Dialog Box* section for more information. If you change the ohmmeter connections, re-activate the circuit to get a reading.

Decibels—Measures decibel voltage loss between two nodes in a circuit. To measure the decibels, select this option and attach the multimeter's probes in parallel with the load, as shown below.



- The decibel standard for calculating dB is preset to 774.597 mV, but can be changed by clicking **Set**. Refer to the *Internal Settings - Multimeter Settings Dialog Box* section for more information. Decibel loss is calculated as follows:

$$dB = 20 * \log 10\left(\frac{V_{out}}{V_{in}}\right)$$

Signal Mode (AC or DC)

The **Sine-wave** button measures the root-mean-square (RMS) voltage or current of an AC signal. Any DC component of the signal will be eliminated, so only the AC component of the signal is measured.

The **DC** button measures the current or voltage value of a DC signal.

Note To measure the RMS voltage of a circuit with both AC and DC components, connect an AC voltmeter as well as a "DC" voltmeter across the appropriate nodes and measure the AC and DC voltage.

The following formula can be used to calculate RMS voltage when both AC and DC components are in the circuit. This is not a universal formula and should be used in conjunction with Multisim only.

$$\text{RMS voltage} = \sqrt{\left(V_{dc}^{2} + V_{ac}^{2}\right)}$$

Internal Settings - Multimeter Settings Dialog Box

Ideal meters have no effect on the circuit being measured. An ideal voltmeter would have infinite resistance, so no current could flow through it while it is attached to a circuit. An ideal ammeter would present no resistance to a circuit. Real meters do not achieve this ideal, so their readings will very closely match theoretical, calculated values for a circuit, but never with absolute precision.

The multimeter in Multisim uses very small and very large numbers that approximate zero and infinity to calculate near-ideal values for the circuit. For special cases, however, the meter's behavior can be changed by changing these values used to model its effect on the circuit. (The values must be higher than 0.)

For example, if testing the voltage of a circuit with very high resistance, increase the voltmeter's resistance. If measuring the current of a circuit with very low resistance, decrease the ammeter's resistance even further.

Note Very low ammeter resistance in a high-resistance circuit may result in a mathematical roundoff error.

Complete the following steps to display the default internal settings:

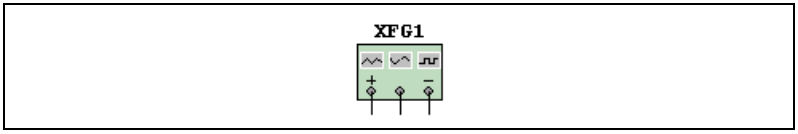
1. Click **Set**. The **Multimeter Settings** dialog box appears.
2. Change the desired options.
3. To save your changes, click **OK**. To cancel them, click **Cancel**.

Function Generator

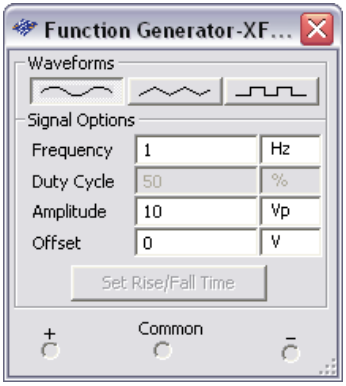
The **Function Generator** is a voltage source that supplies sine, triangular or square waves. It provides a convenient and realistic way to supply stimulus signals to a circuit. The waveform can be changed and its frequency, amplitude, duty cycle and DC offset can be controlled. The frequency range is great enough to produce conventional AC as well as audio- and radio-frequency signals.

To place the instrument, click on the **Function Generator** button in the **Instruments** toolbar and click to place its icon on the workspace. The icon is used to wire the function generator to the circuit.

The function generator has three terminals through which waveforms can be applied to a circuit. The common (center) terminal provides a reference level for the signal.



Double-click on the icon (shown above) to open the instrument face (shown below), which is used to enter settings and view measurements.



To reference a signal from ground, connect the **Common** terminal to the ground component.

The positive terminal (**+**) provides a waveform in the positive direction from the neutral common terminal. The negative terminal (**–**) provides a waveform in the negative direction.

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Function Generator Settings

Waveform Selection

You can select one of three different types of waveforms as the output.

To select the waveform, click the **Sine-**, **Triangular-** or **Square-wave** button.

Complete the following steps to set the rise/fall time parameters for square waves:

1. Click the **Square-wave** button. The **Set Rise/Fall Time** button becomes active.
2. Click the **Set Rise/Fall Time** button to display the **Set Rise/Fall Time** dialog box.
3. Enter the desired **Rise/Fall Time** and click **Accept**.

Signal Options

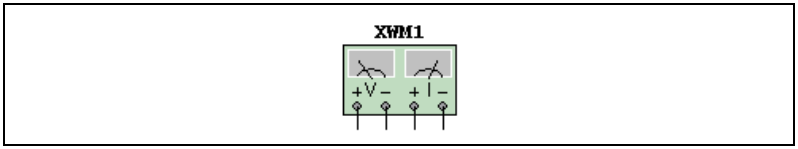
The available **Signal Options** depend on the chosen waveform:

- **Frequency** (1Hz—999 MHz)—The number of cycles per second of the generated signal.
- **Duty Cycle** (1%—99%)—The ratio of on-period to off-period of a triangular or square wave. This option is disabled for sine waves.
- **Amplitude** (1mV—999 kV)—Controls the signal's voltage, measured from its DC level to its peak. If the leads are connected to the common and either the positive or the negative terminal, the wave's peak-to-peak measurement is twice its amplitude. If the output comes from the positive and negative terminals, the wave's peak-to-peak measurement is four times its amplitude.
- **Offset** (–999 kV and 999 kV)—Controls the DC level about which the alternating signal varies. An offset of 0 positions the waveform along the oscilloscope's x-axis (provided its Y POS setting is O). A positive value shifts the DC level upward, while a negative value shifts it downward. Offset uses the units set for **Amplitude**.

Wattmeter

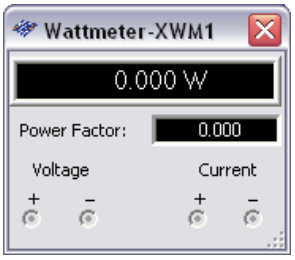
The **Wattmeter** measures power. It is used to measure the magnitude of the active power, that is, the product of the voltage difference and the current flowing through the current terminals in a circuit. The results are shown in Watts. The **Wattmeter** also displays the power factor, calculated by measuring the difference between the voltages and the current, and multiplying them together. The power factor is the cosine of the phase angle before the voltage and current.

To use the instrument, click on the **Wattmeter** button in the **Instruments** toolbar and click to place its icon on the workspace. The icon, shown below, is used to wire the wattmeter to the circuit.



Double-click on the icon to open the instrument face. This is where you enter settings and view measurements.

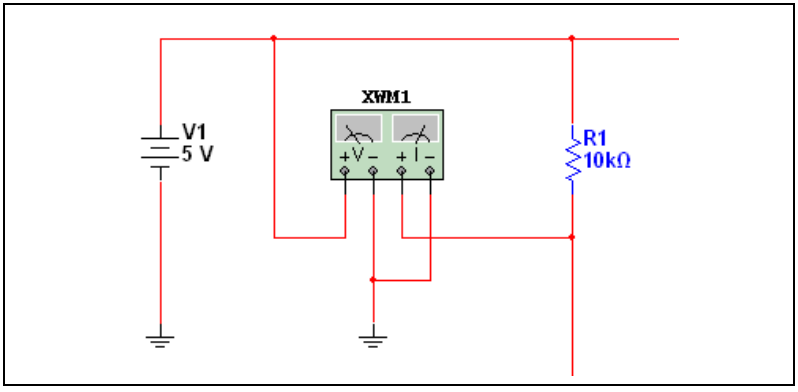
The field at the top of the instrument face (shown below) displays the average power. The **Power Factor** field displays the power factor between 0 and 1.



Connecting the Wattmeter

Connect the **V** terminals in series with the load, and the **I** terminals in parallel with the load.

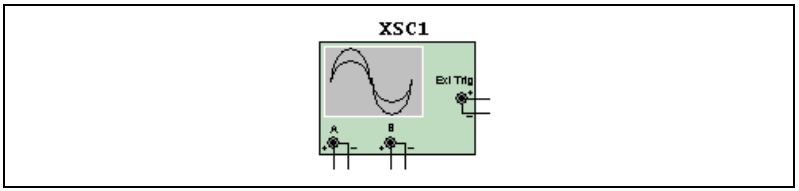
An example of a connected wattmeter is shown below.



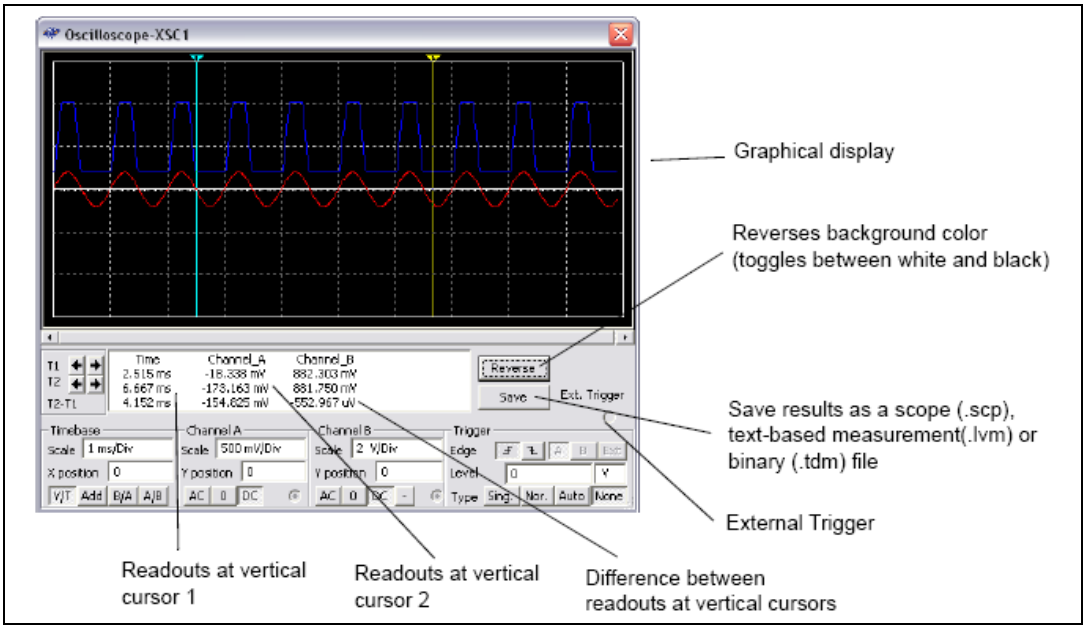
Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Oscilloscope

To place the instrument, click on the **Oscilloscope** button in the **Instruments** toolbar and click to place its icon (shown below) on the workspace. The icon is used to wire the oscilloscope to the circuit.



Double-click on the icon to open the instrument face (shown below), which is used to enter settings and view measurements.



The dual-channel oscilloscope displays the magnitude and frequency variations of electronic signals. It can provide a graph of the strength of one or two signals over time, or allow comparison of one waveform to another.

Note If you choose to save the results as either `.lvm` or `.tdm` files, the **Data resampling settings** dialog box displays. Refer to the *Saving Files* section of Chapter 10, *Analyses*, for more information on this dialog. Along with the oscilloscope's **Save** button, you can also save simulation results in the **Grapher** window. Refer to the *Saving Files* section of Chapter 10, *Analyses*, for more information.

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Oscilloscope Settings

Timebase

The time base setting controls the scale of the oscilloscope's horizontal or x-axis when comparing magnitude against time (Y/T). **Scale** is the value of the divisions on the x-axis. To get a readable display, adjust the **Scale** in inverse proportion to the frequency setting on the function generator or AC source—the higher the frequency, the lower (or more magnified) the time base. For example, if you want to see one cycle of a 1 kHz signal, the time base should be around 1 millisecond.

X position

This setting controls the signal's starting point on the x-axis. When **X position** is 0, the signal starts at the left edge of the display. A positive value (for example, 2.00) shifts the starting point to the right. A negative value (for example, -3.00) shifts the starting point to the left.

Axes (Y/T, B/A, and A/B)

The axes of the oscilloscope display can be switched from showing waveform magnitude against time (**Y/T**) to showing one input channel against the other (**A/B** or **B/A**). The latter settings display frequency and phase shifts, known as Lissajous patterns, or they can display a hysteresis loop. When comparing channel A's input against channel B's (**A/B**), the scale of the x-axis is determined by the volts-per-division setting for channel B (and vice versa).

Add

The **Add** button adds traces A and B.

Grounding

It is not necessary to ground the oscilloscope, as long as the circuit to which it is attached is grounded.

Channel A and Channel B Settings

Scale

This setting determines the **Scale** of the y-axis. It also controls the x-axis scale when **A/B** or **B/A** is selected.

To get a readable display, adjust the scale in relation to the channel's expected voltage. For example, an input AC signal of 3 volts fills the oscilloscope's display vertically when the y-axis is set to 1 V/Div. If the **Scale** setting is increased, the waveform becomes smaller. If the **Scale** setting is decreased, the waveform's top is cut off (clipped).

Y position

This setting controls the point of origin for the y-axis. When **Y position** is set to 0.00, the point of origin is the intersection with the x-axis. Increasing **Y position** to 1.00, for example, moves 0 (the point of origin) up to the first division above the x-axis. Decreasing **Y position** to –1.00 moves 0 down to the first division below the x-axis.

Changing the **Y position** setting for channels A and B may help distinguish their waveforms for comparison.

Input Coupling (AC, 0, DC, and -)

With **AC** coupling, only the AC component of a signal is displayed. AC coupling has the effect of placing a capacitor in series with the oscilloscope's probe. As on a real oscilloscope using AC coupling, the first cycle displayed is inaccurate. Once the signal's DC component has been calculated and eliminated during the first cycle, the waveforms will be accurate.

With **DC** coupling, the sum of the AC and DC components of the signal is displayed. **0** displays a reference flat line at the point of origin set by **Y position**.

"**–**" inverts the trace 180 degrees. When used in conjunction with the timebase **Add** button, it subtracts Trace B from Trace A.

Caution Do not place a coupling capacitor in series with an oscilloscope probe. The oscilloscope will not provide a path for current, and the analysis will consider the capacitor improperly connected. Instead, choose AC coupling.

Trigger

These settings determine the conditions under which a waveform is first displayed on the oscilloscope.

Trigger Edge

To start displaying the waveform on its positive slope or rising signal, click the **Ascending Edge** button.

To start with the negative slope or falling signal, select the **Descending Edge** button.

Trigger Level

The trigger **Level** is the point on the oscilloscope's y-axis that must be crossed by the waveform before it is displayed.

Tip A flat waveform will not cross the trigger level. To see a flat signal, make sure the triggering signal is set to **Auto**.

Trigger Type

Triggering can be internal, with reference to the input signal for channel A or B, or external, with reference to a signal through the external trigger terminal. If a flat signal is expected, or if signals are to be displayed as soon as possible, select **Auto**. The oscilloscope will wait for two pages (that is, full-width displays of the oscilloscope) for a trigger before forcing a trigger.

Select the **Sing.** button to make the oscilloscope trigger a single pass when the trigger level is met. Once the trace gets to the end of the scope face, the trace will not change until you click **Sing.** again.

Select the **Nor.** button to make the oscilloscope refresh every time the trigger level is met.

Select the **None** button if you do not wish to set triggering.

Viewing Oscilloscope Results

Using Cursors and Readouts

To display the exact values of the wave, drag the vertical cursor until the desired value appears. You can also move a cursor to a precise location by right-clicking on it and using the pop-up menu that appears. Refer to the *Cursor Pop-up Menu* section of Chapter 10, *Analyses*, for more information.

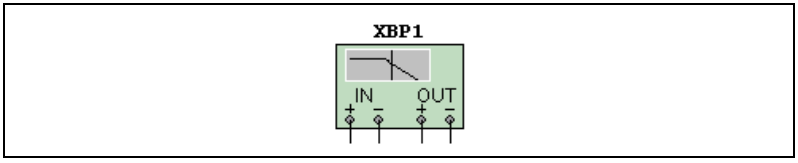
The box below the display shows the time and the voltage at the probe connections, where the vertical cursor intersects the sine wave, and the difference between the two positions.

Once a circuit has been activated and its behavior simulated, you may move the oscilloscope's probes to other nodes without re-activating the circuit. Moving the probes automatically redraws the waveforms for the new nodes. If you fine-tune the oscilloscope's settings either during or after simulation, the display redraws automatically.

Note If the oscilloscope settings or analysis options are changed to provide more detail, the waveforms may appear choppy or uneven. If so, activate the circuit again to get more detail. You can also increase the precision of a waveform by increasing the simulation time step. Refer to the *Interactive Simulation Settings* section for more information.

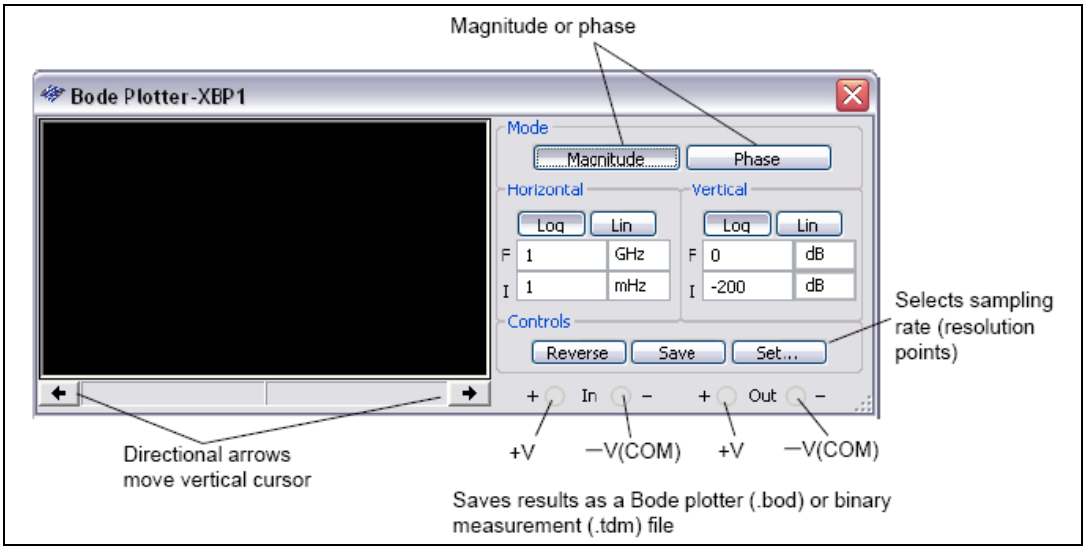
Bode Plotter

To place the instrument, click the **Bode Plotter** button in the **Instruments** toolbar and click to place its icon on the workspace. The icon (shown below) is used to wire the Bode plotter to the circuit.



The Bode plotter produces a graph of a circuit's frequency response and is most useful for analyzing filter circuits. The Bode plotter is used to measure a signal's voltage gain or phase shift. When the Bode plotter is attached to a circuit, a spectrum analysis is performed.

Double-click on the icon to open the instrument face (shown below), which is used to enter settings and view measurements.



Note Along with the Bode plotter's **Save** button, you can also save simulation results in the **Grapher** window. Refer to the *Opening Files* section of Chapter 10, *Analyses*, for more information.

The initial and final values of the vertical and horizontal scales are preset to their maximum value. These values can be changed to see the plot on a different scale. If the scale is expanded or the base changed after simulation is complete, you may need to activate the circuit again to get more detail in the plot. Unlike most test instruments, if the Bode plotter's probes are moved to different nodes, it is necessary to re-activate the circuit to ensure accurate results.

Caution The Bode plotter does not work in conjunction with the **Function Generator** with a differential setting (that is, when using the + and - terminals). You can only use the Bode plotter with the **Function Generator** if it is wired as an absolute generator. If you wish to use the Bode plotter with differential settings, you must use an AC source.

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Bode Plotter Settings

Resolution Points - Settings Dialog Dialog Box

Complete the following steps to set the resolution of your Bode plot:

1. Click **Set** to display the **Settings Dialog**.
2. Enter the desired number of **Resolution Points** and click **Accept**.

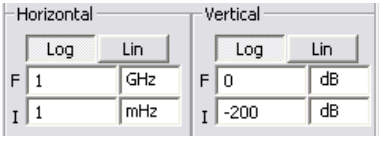
Magnitude or Phase

Magnitude measures the ratio of magnitudes (voltage gain, in decibels) between two nodes, V+ and V–. **Phase** measures the phase shift (in degrees) between two nodes. Both are plotted against frequency (in Hz).

Complete the following steps to V+ and V– are single points in a circuit:

1. Attach the positive IN terminal and the positive OUT terminal to connectors at V+ and V–.
2. Attach the negative IN and OUT terminals to a ground component.
3. If V+ (or V–) is the magnitude or phase across a component, attach both IN terminals (or both OUT terminals) on either side of the component.

Vertical and Horizontal Axis Settings



Base Settings

A logarithmic base is used when the values being compared have a large range, as is generally the case when analyzing frequency response. For example, if measuring a signal's voltage gain, the decibel value is calculated as follows:

$$dB = 20 * \log 10\left(\frac{V_{out}}{V_{in}}\right)$$

The base scale can be changed from logarithmic (**Log**) to linear (**Lin**) without the circuit being activated again. (Only when using a logarithmic scale is the resulting graph referred to as a Bode plot).

Horizontal Axis Scale

The horizontal or x-axis always shows frequency. Its scale is determined by the initial (**I**) and final (**F**) settings for the horizontal axis. Since a frequency response analysis requires a large frequency range, a logarithmic scale is often used.

Note When setting the horizontal axis scale, the initial (**I**) frequency must be less than the final (**F**) frequency.

Vertical Axis Scale

The units and scale for the vertical axis depend on what is being measured and the base being used, as shown in the table below.

When Measuring...	Using the Base...	Minimum Initial Value is...	Maximum Final Value is...
Magnitude (gain)	Logarithmic	–200 dB	200 dB
Magnitude (gain)	Linear	0	10e+09
Phase	Linear	–720°	–720°

When measuring voltage gain, the vertical axis shows the ratio of the circuit's output voltage to its input voltage. For a logarithmic base, the units are decibels. When measuring phase, the vertical axis always shows the phase angle in degrees. Regardless of the units, you can set initial (**I**) and final (**F**) values for the axis using the Bode plotter's controls.

Viewing Bode Plotter Results

Move the Bode plotter's vertical cursor to get a readout of the frequency and magnitude or phase at any point on the plot. The vertical cursor is stored at the left edge of the Bode plotter display.

Complete the following steps to move the vertical cursor:

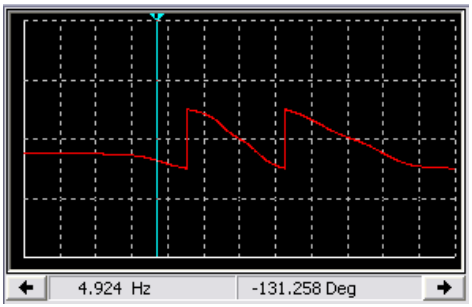
1. Click the arrows near the bottom of the Bode plotter.

   Or

   Drag the vertical cursor from the left edge of the Bode plotter display to the point on the plot you want to measure.

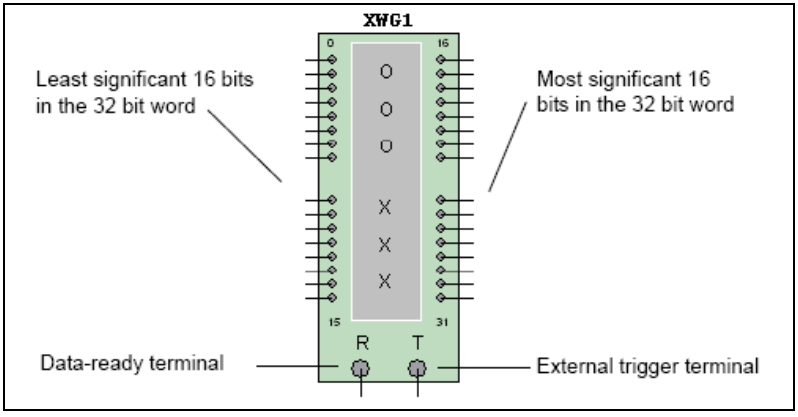
Note You can also move the cursor by right-clicking on it and using the pop-up menu that appears. Refer to the *Cursor Pop-up Menu* section of Chapter 10, *Analyses*, for more information. This feature is useful when locating the –3 dB point on a Bode plot.

The magnitude (or phase) and frequency at the intersection of the vertical cursor and the trace are shown in the boxes below the display, as in the example below.



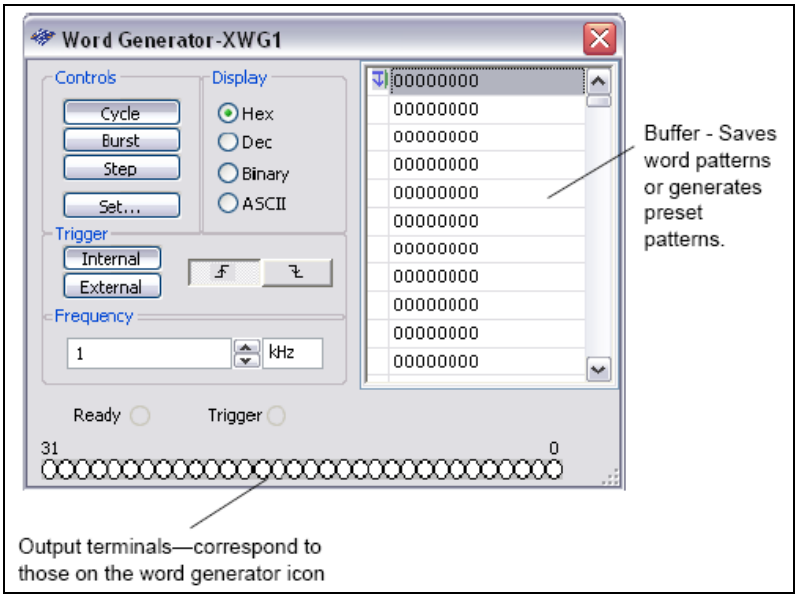
Word Generator

To place the instrument, click on the **Word Generator** button in the **Instruments** toolbar and click to place its icon on the workspace. The icon is used to wire the word generator to the circuit.



Use the word generator to send digital words or patterns of bits into circuits to provide stimulus to digital circuits.

Double-click on the icon to open the instrument face, which is used to enter settings and view measurements.



Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Word Generator Settings

Entering Words

The right side of the word generator instrument face displays rows of numbers, ranging from 00000000 to FFFFFFFF in hexadecimal (0 to 4,294,967,295 in decimal). The type of number that displays can be **Hex**, **Dec**, **Binary** or **ASCII**, depending on the button selected in the **Display** box. Each horizontal row represents one word. When the word generator is activated, a row of bits is sent in parallel to the corresponding terminals at the bottom of the instrument.

To change a bit value in the word generator, select the number you want to modify and type the new value in its field using the appropriate number format.

As the words are transmitted by the word generator, the value of each bit appears in the circles representing the output terminals at the bottom of the instrument.

Controls

To inject the 32-bit words into a circuit, click **Step**, **Burst** or **Cycle**.

To transmit one word at a time into the circuit, click **Step**.

To send all words in sequence, click **Burst**. Click **Cycle** to send a continuous stream of words that can be stopped by clicking the **Stop simulation** button on the **Simulation** toolbar.

Use a breakpoint when you want to pause the stream of words at a specified word.

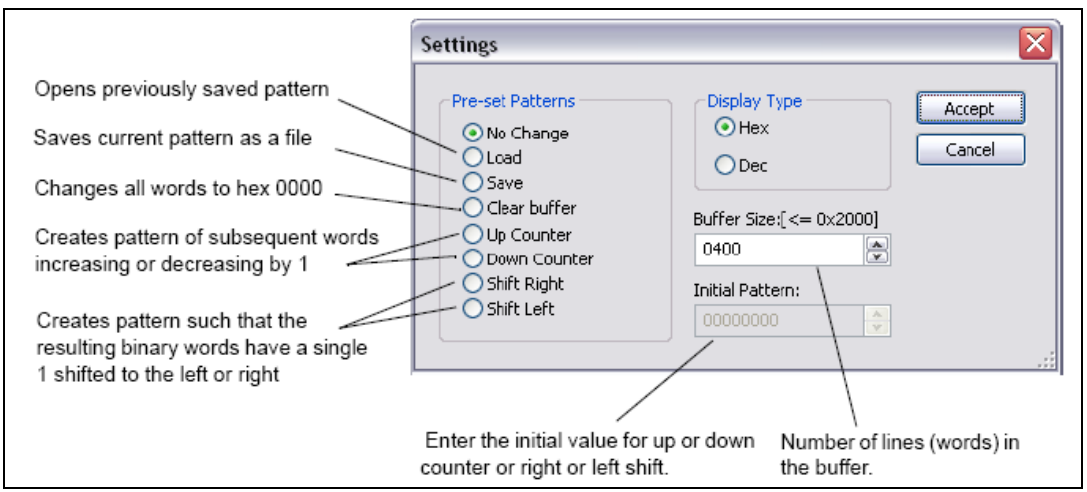
To insert a breakpoint, select the word in the buffer scroll list where you want the input to stop, right-click and select **Set Breakpoint** from the pop-up that appears. A "stop sign" marks a breakpoint in the scroll list.

To remove a breakpoint, right-click on an existing breakpoint in the buffer scroll list and select **Delete Breakpoint** from the pop-up that appears.

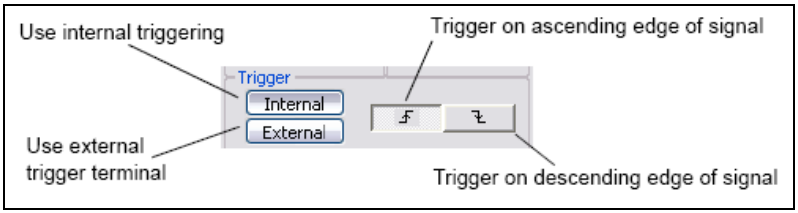
More than one breakpoint can be used. Breakpoints affect both **Cycle** and **Burst**.

Settings Dialog Box

Click **Set** to display a set of options that allow you to save word patterns entered in the word generator to a file and load previously saved word patterns. This function can also be used to generate useful patterns or to clear the display.



Triggering



To use the word generator clock to trigger input from the word generator's input field to the circuit, click **Internal**. To use input through the external terminal instead, with each input cycle causing one word to be transmitted, click **External**.

Use the "ascending/descending edge" buttons to control whether the input signal triggers the word generator on its ascending or descending edge.

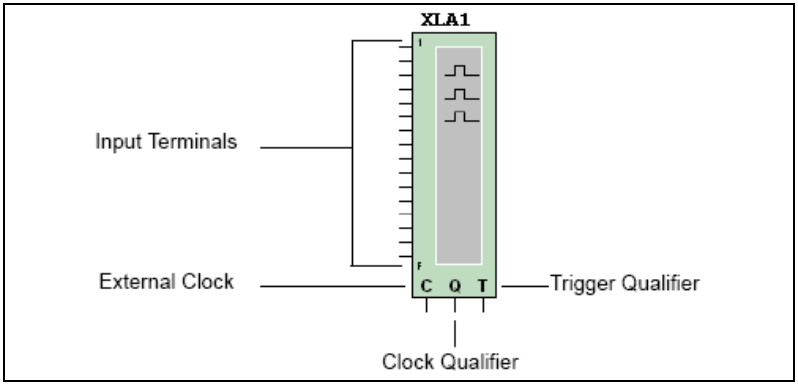
Frequency and Data Ready

Set the clock **Frequency** of the word generator in Hz, kHz or MHz. Each word is placed on the output terminals for the duration of one clock cycle.

Ready—Lets the circuit know that data from the word generator is ready.

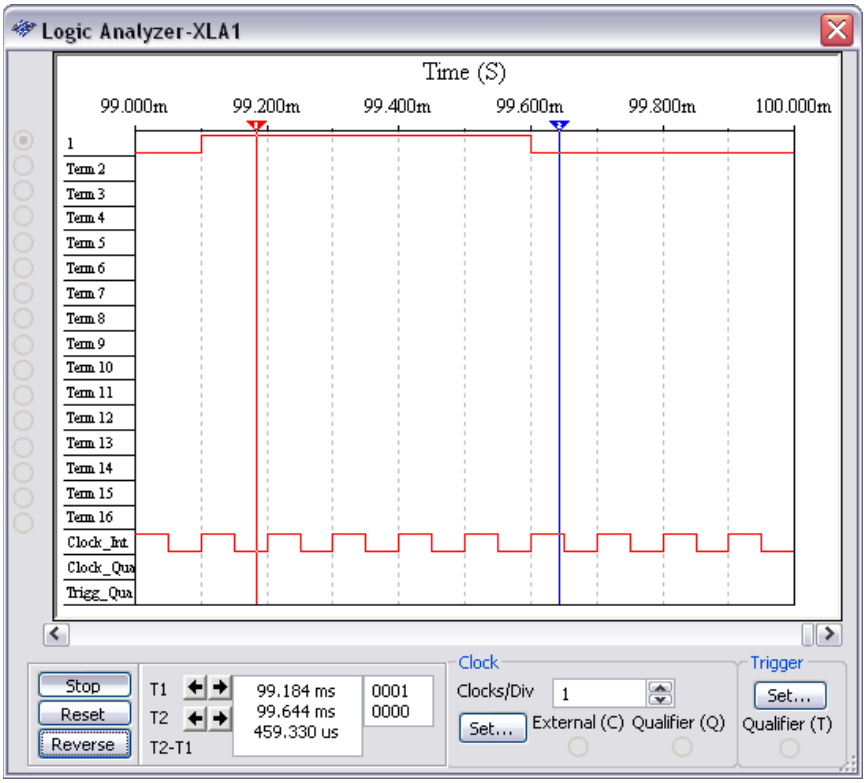
Logic Analyzer

To place the instrument, click on the **Logic Analyzer** button in the **Instruments** toolbar and click to place its icon on the workspace. The icon is used to wire the logic analyzer to the circuit.



The logic analyzer displays the levels of up to 16 digital signals in a circuit. It is used for fast data acquisition of logic states and advanced timing analysis to help design large systems and carry out troubleshooting.

Double-click on the icon to open the instrument face, which is used to enter settings and view measurements.



The 16 circles on the left side of the icon correspond to the terminals and horizontal rows across the instrument face. When the terminal is connected with a node, its circle is displayed with a black dot and the node's name and color are displayed. Otherwise the terminal circle is displayed without a black dot.

When a circuit is activated, the logic analyzer records the input values on its terminals. When the triggering signal is seen, the logic analyzer displays the pre- and post-trigger data. Data is displayed as square waves over time. The top row displays values for channel 1, the next row displays values for channel 2, and so on. The binary value of each bit in the current word is displayed in the terminals on the left side of the instrument face. The time

axis is displayed as the top axis of the signal display dialog box. The dialog box also displays the internal clock signal, external clock signal, external clock qualify signal and trigger qualify signal.

To specify the number of samples stored before and after triggering, click **Set** in the **Clock** box or use the default instrument settings. Refer to the *Interactive Simulation Settings* section for more information.

The logic analyzer stores data until it reaches the pre-trigger number of samples. Then, it begins discarding samples as new samples appear until it sees the trigger signal. After the trigger signal, samples are stored up to the value of the post-trigger samples.

The time position automatically displays the time position values of the two crosshair cursors, T1 and T2, when sampling is stopped. It also automatically moves the first crosshair cursor T1 to the position of time zero, when sampling is stopped.

To change the threshold voltage, use the default instrument settings. Refer to the *Interactive Simulation Settings* section for more information.

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Logic Analyzer Settings

Stop & Reset

Selecting **Simulate»Pause** while the circuit is simulating, pauses both the circuit and the logic analyzer. By using the scroll bar at the bottom of the instrument face, you can view the results of the logic analyzer at any period from the first instance the trigger condition was met as long as the logic analyzer was not reset. If you did reset the logic analyzer, you would be able to view the results from the time the instrument was reset and the trigger condition is met.

To restart the instrument, click **Reset**. This action will clear all information in the instrument and start storing new information from the instance the trigger condition is met after you clicked **Reset**.

To discard stored data when the logic analyzer is not triggered, click **Stop**. If the logic analyzer is already triggered and displaying data, clicking **Stop** will stop the instrument and allow the simulation to continue. After you have stopped the instrument, you must reset the logic analyzer to begin storing data again.

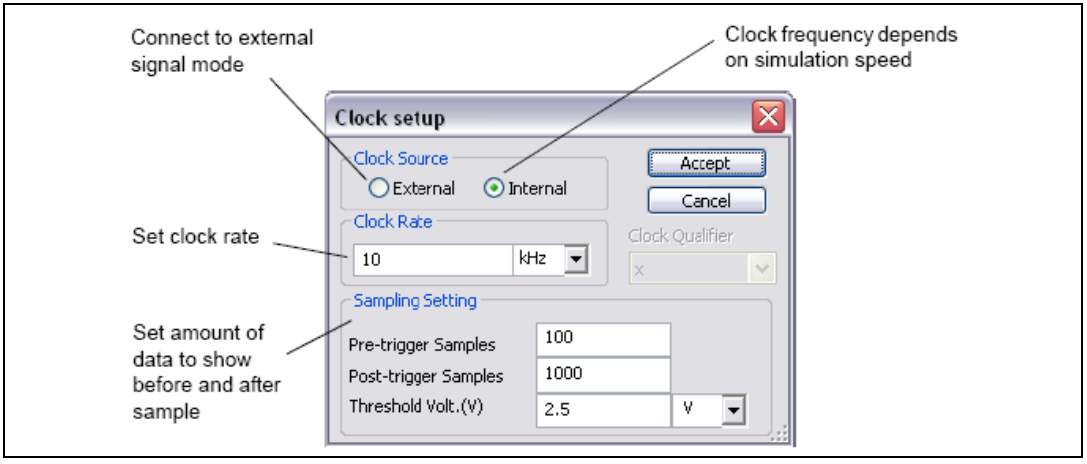
To clear the logic analyzer's display, click **Stop**, then **Reset**. Clearing the display does not disrupt the simulation. The instrument will start storing data from the instance the trigger condition is met after you click **Reset**.

Clock Setup

The clock informs the logic analyzer when to read an input sample. The clock can be internal or external.

Complete the following steps to adjust the clock settings:

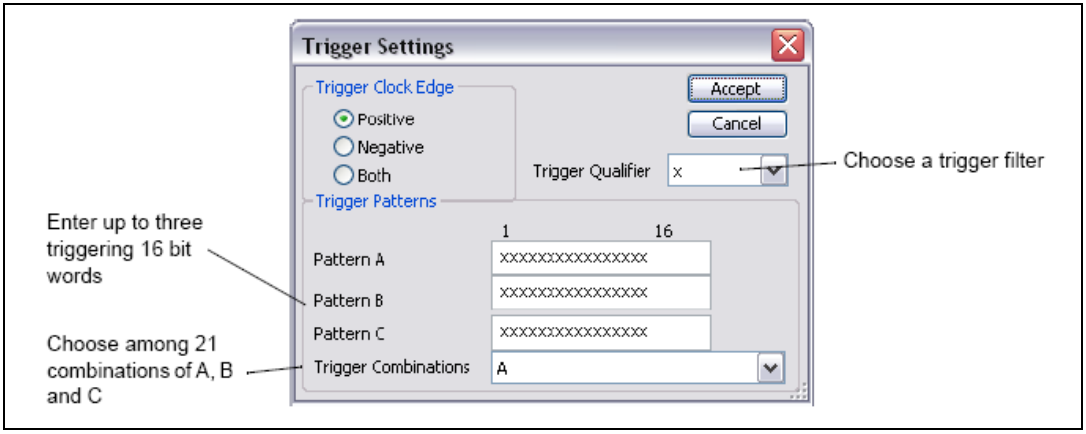
1. Click **Set** in the **Clock** area of the logic analyzer. The **Clock Setup** dialog box appears.



2. Select external or internal clock mode.
3. Set the internal clock rate. The clock qualifier is an input signal that filters the clock signal. If it is set to "x", then the qualifier is disabled and the clock signal determines when samples are read. If it is set to "1" or "0", the samples are read only when the clock signal matches the selected qualifier signal.
4. Set how much data to show before (**Pre-trigger samples**) and after (**Post-trigger samples**) the sample.
5. Click **Accept**.

Trigger Settings

The logic analyzer can be made to trigger upon reading a specified word or combination of words or when meeting the increase edge or decrease edge of the clock signal.



Complete the following steps to specify up to three trigger words or word combinations:

1. Click **Set** in the **Trigger** box of the logic analyzer.
2. Select **Positive**, **Negative** or **Both** positive and negative clock edge.
3. Click in the box labeled **Pattern A**, **Pattern B**, or **Pattern C** and enter a binary word. An “x” means either 1 or 0.
4. From the **Trigger Combinations** drop-down list, select the desired combination. (See below for a list of combinations.)
5. From the **Trigger Qualifier** drop-down list, select the desired trigger qualifier. The trigger qualifier is an input signal that filters the triggering signal. If it is set to “x”, then the qualifier is disabled and the trigger signal determines when the logic analyzer is triggered. If it is set to “1” or “0”, the logic analyzer is triggered only when the triggering signal matches the selected trigger qualifier.
6. Click **Accept**.

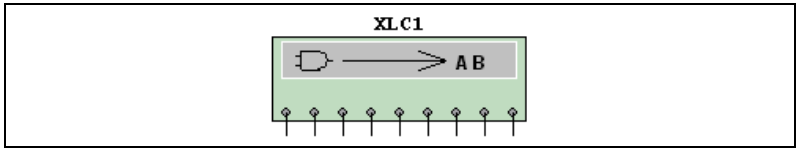
The possible trigger combinations are:

A	B	C
A or B	A or C	B or C
A OR B OR C	A AND B	A AND C
B AND C	A AND B AND C	A NOT B
A NOT C	B NOT C	A THEN B
A THEN C	B THEN C	(A OR B) THEN C
A THEN (B OR C)	A THEN B THEN C	A THEN (B WITHOUT C)

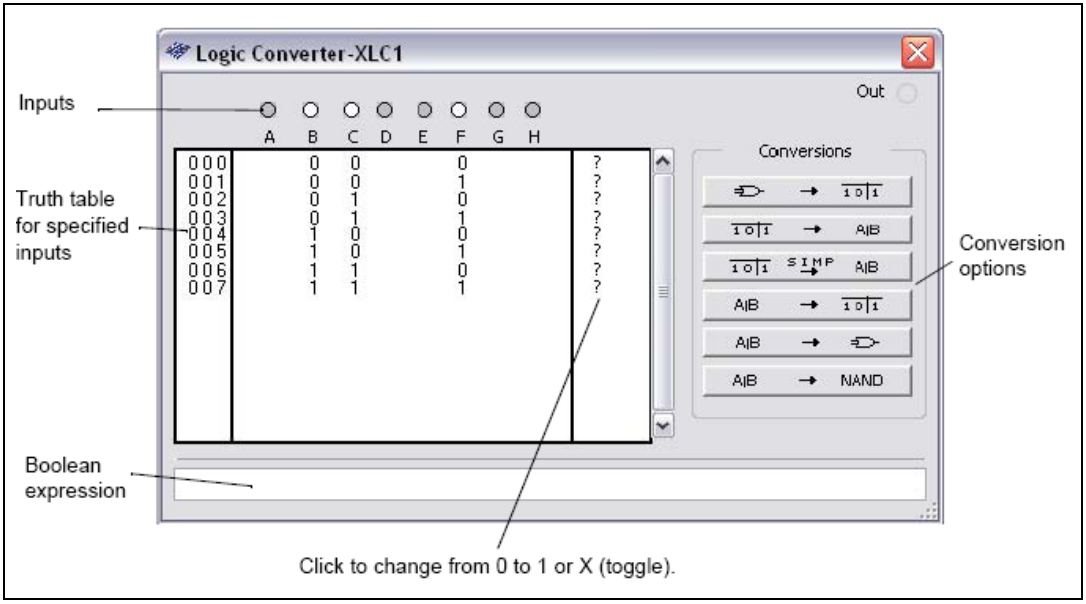
Logic Converter

To place the instrument, click on the **Logic Converter** button in the **Instruments** toolbar and click to place its icon on the workspace. The icon is used to wire the logic converter to the circuit.

The logic converter is able to perform several transformations of a circuit representation or digital signal. This is a useful tool for digital circuit analysis, but has no real-world counterpart. It can be attached to a circuit to derive the truth table or Boolean expression the circuit embodies, or to produce a circuit from a truth table or Boolean expression.



Double-click on the icon to open the instrument face, which is used to enter settings and view measurements.



Click the circles or the labels below them to display the inputs for that terminal.

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Logic Converter Settings

The following table describes the buttons found in the **Logic Converter**:

Button	Description
	Circuit to Truth Table button. Click to derive a truth table from a circuit.
	Truth Table to Boolean Expression button. Click on this button to convert a truth table to a boolean expression.
	Simplify button. Click to convert a truth table to a simplified boolean expression, or to simplify an existing boolean expression.

Button	Description
	Boolean Expression to Truth Table button. Click to convert a boolean expression to a truth table.
	Boolean Expression to Circuit button. Click to convert a boolean expression to a circuit.
	Boolean Expression to NAND button. Click to convert a boolean expression to a circuit built using only NAND gates.

Deriving a Truth Table from a Circuit

Complete the following steps to derive a truth table from a circuit schematic:

1. Attach the input terminals of the logic converter to up to eight nodes in the circuit.
2. Connect the single output of the circuit to the output terminal in the **logic converter** icon.
3. Click the **Circuit to Truth Table** button. The truth table for the circuit appears in the logic converter's display.

Entering and Converting a Truth Table

Complete the following steps to construct a truth table:

1. Click the input channels you want, from A to H, across the top of the logic converter. The display area below the terminals fills up with the necessary combinations of ones and zeros to fulfill the input conditions. The values in the output column on the right are initially set to "?".
2. Edit the output column to specify the desired output for each input condition. To change an output value, click on it to switch among the three possible settings: "0", "1" and "X" (an "X" indicates that either 1 or 0 is acceptable).

To convert a truth table to a Boolean expression, click the **Truth Table to Boolean Expression** button. The Boolean expression is displayed at the bottom of the logic converter.

To convert a truth table to a simplified Boolean expression, or to simplify an existing Boolean expression, click the **Simplify** button.

The simplification is performed by the Quine-McCluskey method, rather than the more familiar Karnaugh mapping technique. Karnaugh mapping works for only small numbers of variables and requires human intuition, while Quine-McCluskey has proved to be exhaustive for any number of variables but is too cumbersome for manual solutions.

Note Simplifying a Boolean expression requires substantial memory. If sufficient memory is not available, Multisim may not be able to complete this operation.

Entering and Converting a Boolean Expression

A Boolean expression can be entered in the box at the bottom of the logic converter using either sum-of-products or product-of-sums notation.

To convert a Boolean expression to a truth table, click the **Boolean Expression to Truth Table** button.

To convert a Boolean expression to a circuit, click the **Boolean Expression to Circuit** button.

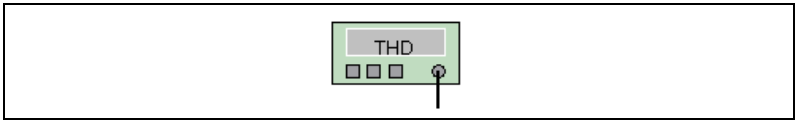
The logic gates that fulfill the Boolean expression appear on the circuit window.

To see a circuit that fulfills the conditions of the Boolean expression using only NAND gates, click the **Boolean Expression to NAND** button.

Note The width of the wire in the resulting circuit is determined by the **Wire Width** setting in the **Wiring** tab of the **Sheet Properties** dialog box.

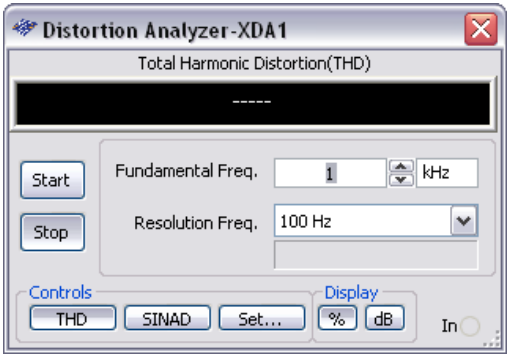
Distortion Analyzer

To place the instrument, click on the **Distortion Analyzer** button in the **Instruments** toolbar and click to place its icon on the workspace. The icon is used to wire the distortion analyzer to the circuit.



A typical distortion analyzer provides distortion measurements for signals in the range of 20 Hz to 100 kHz, including audio signals.

Double-click on the icon to open the instrument face, which is used to enter settings and view measurements.



The types of measurements performed are either Total Harmonic Distortion (**THD**) or Signal Plus Noise and Distortion (**SINAD**).

To set the way results are to be displayed for either type of measurement, click **Set** to display the **Settings** dialog box.

Note The **THD Definition** box is where you set the definition used to calculate THD (**IEEE** defines this slightly differently from **ANSI/IEC**).

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Distortion Analyzer Settings

Harmonic Distortion

Harmonic distortion produces signals at harmonics of the test frequency. For example, for a 1 kHz signal, the harmonics may be at 2 kHz, 3 kHz, and 4 kHz.

A very sharp tunable notch is required to measure harmonic distortion. This is tuned to the test frequency such as 1 kHz, which will remove the 1 kHz signal, leaving only the harmonics or the distortion. The distortion harmonics are measured and the resulting value is compared to the amplitude of the test signal.

SINAD

This type of measurement measures the ratio of (signal plus noise and distortion)/(noise and distortion).

Spectrum Analyzer

The **Spectrum Analyzer** is used to measure amplitude versus frequency. It performs a similar function in the frequency domain as an oscilloscope performs in the time domain. It operates by sweeping through a range of frequencies. The amplitude of the signal at the input of the receiver is plotted against the frequency of the signal. This instrument is capable of measuring a signal's power at various frequencies, and helps determine the existence of the frequency components' signal.

The **Spectrum Analyzer** is part of the RF Design module. Refer to the *RF Instruments* section of Chapter 14, *RF*, for more information.

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Network Analyzer

The **Network Analyzer** is used to measure the scattering parameters (or S-parameters) of a circuit, commonly used to characterize a circuit intended to operate at higher frequencies. These S-parameters are used to derive matching cells using other Multisim analyses. The **Network Analyzer** also calculates H, Y, Z parameters.

The circuit is idealized as a two-port network. To properly use the network analyzer, the circuit must be left open at its input and output ports. During simulation the network analyzer completes the circuit being analyzed by inserting its subcircuits. You need to remove these subcircuits from the circuit before performing other analysis and simulation.

The **Network Analyzer** is part of the RF Design module. Refer to the *RF Instruments* section of Chapter 14, *RF*, for more information.

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Measurement Probe

Using the **Measurement Probe** is a fast and easy way to check voltage, current and frequency readings at different points in the circuit.

The **Measurement Probe** can be used as a:

- **Dynamic Probe**—During simulation, drag the probe to point to any wire to obtain an "on-the-fly" reading.

  The probe displays the following:

  – **V**—Instantaneous voltage
  – **V(p-p)**—Peak-to-peak voltage
  – **V(rms)**—RMS voltage
  – **V(dc)**—DC voltage
  – **Freq**—Frequency

- **Static Probe**—Multiple probes can be connected to points in the circuit before or during simulation. These probes remain stationary, and will contain the data from the simulation until another simulation is run, or the data is cleared. In addition to the various voltage readings and the frequency reading found in dynamic probes, static probes can also include current readings. Static probes are listed in analyses' **Output** tabs in the **Variables in circuit** list. Refer to the *Output Tab* section of Chapter 10, *Analyses*, for more information.

Note Dynamic probes cannot measure current; static probes placed after simulation has run also cannot measure current. This is because the SPICE netlist must be modified in order to measure current; this is done when the simulation is restarted.

Measurement Probe Settings

This section contains instructions on how to set the properties for both dynamic and static probes.

Dynamic Probe Settings

Complete the following steps to set up the dynamic probe properties:

1. Select **Simulate»Dynamic Probe Properties** to display the **Probe Properties** dialog box.
2. Click the **Display** tab and set the following in the **Color** box as desired:
   - **Background**—The currently selected color for the probe's text window background displays here. Select either **System (Tooltip)**

or **Custom**. For **Custom**, click **Select Color** and choose the desired background color from the **Colors** dialog that appears.

- **Text**—The currently selected color for the text that appears in the probe's text window displays here. Select either **System (Tooltip)** or **Custom**. For **Custom**, click **Select Color** and choose the desired text color from the **Colors** dialog that appears.

Note A "tooltip" is the text that appears attached to the cursor if you hover it above a tool button.

3. In the **Size** box, enter the **Width** and **Height**, or enable **Auto-Resize** to have the information box automatically resize to show all content.
4. Optionally, click on the **Font** tab to change the font used in the probe's text window.
5. Click on the **Parameters** tab.
6. Optionally, enable the **Use reference probe** checkbox and select the desired reference probe from the drop-down list (this is populated by a list of the static probes already in the circuit). Dynamic measurements will be done with reference to the selected reference probe (instead of ground). Using this method, you can for example, display a voltage gain or a phase shift.
7. To hide a parameter (for example, **V(p-p)**), click beside the desired parameter in the **Show** column to toggle to **No**.
8. Use the **Minimum** and **Maximum** columns to set a parameter's range.
9. Optionally, change the number of significant digits for a displayed parameter in the **Precision** column.

Static Probe Settings

Complete the following steps to set up a static (placed) probe's properties:

1. Double-click on the desired probe to display the **Probe Properties** dialog box.
2. Click on the **Display** tab and set the following in the **Color** box as desired:
   - **Background**—The currently selected color for the probe's text window background displays here. Select either **System (Tooltip)** or **Custom**. For **Custom**, click **Select Color** and choose the desired background color from the **Colors** dialog that appears.
   - **Text**—The currently selected color for the text that appears in the probe's text window displays here. Select either **System (Tooltip)**

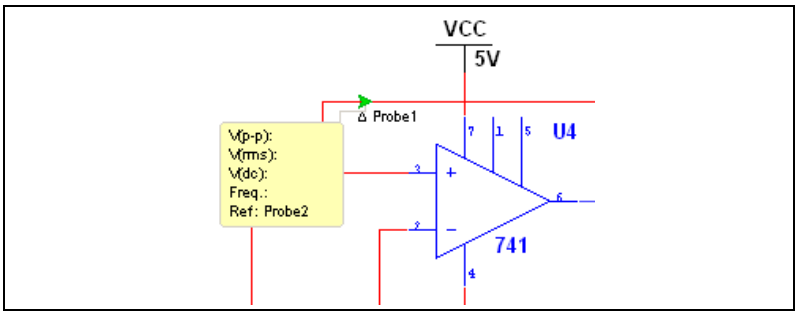
or **Custom**. For **Custom**, click **Select Color** and choose the desired text color from the **Colors** dialog that appears.

Note A "tooltip" is the text that appears attached to the cursor if you hover it above a tool button.

3. In the **Size** box, enter the **Width** and **Height**, or enable **Auto-Resize** to have the information box automatically resize to show all content.
4. Set the following in the **RefDes** box as desired:
   - **RefDes** field—Enter the reference designator for the selected probe. (The default is Probe1, Probe2, etc.)
   - **Hide RefDes**—Hides the reference designator for the selected probe.
   - **Show RefDes**—Shows the reference designator for the selected probe.
   - **Use Global Settings**—Shows or hides the reference designator for the selected probe based on the setting in the **Circuit** tab of the **Sheet Properties** dialog box. Refer to the *Sheet Properties - Circuit Tab* section of Chapter 1, *User Interface*, for more information.
5. Optionally, disable the **Show popup window** checkbox to hide the contents of the selected probe.

Note The **Drawing Layer** list shows the layer on which the probe appears. The default layer is **Static Probe**, but you may select a different layer if desired. Refer to the *Sheet Properties - Visibility Tab* section of Chapter 1, *User Interface*, for more information about layers.

6. Optionally, click on the **Font** tab to change the font used in the probe's text window.
7. Click on the **Parameters** tab.
8. Optionally, enable the **Use reference probe** checkbox and select the desired reference probe from the drop-down list (this is populated by a list of the probes in the circuit, with the exception of the one that is currently selected). Measurements taken at the currently selected probe will be done with reference to the selected reference probe (instead of ground). Using this method, you can, for example, display a voltage gain or a phase shift. If you select this option, a triangle appears beside the probe's reference designator, as shown in the example below, beside Probe1.



Note Reference probes are only for use in interactive simulation. Refer to the *Filtering the Variable Lists* section of Chapter 10, *Analyses*, for information about their use in analyses.

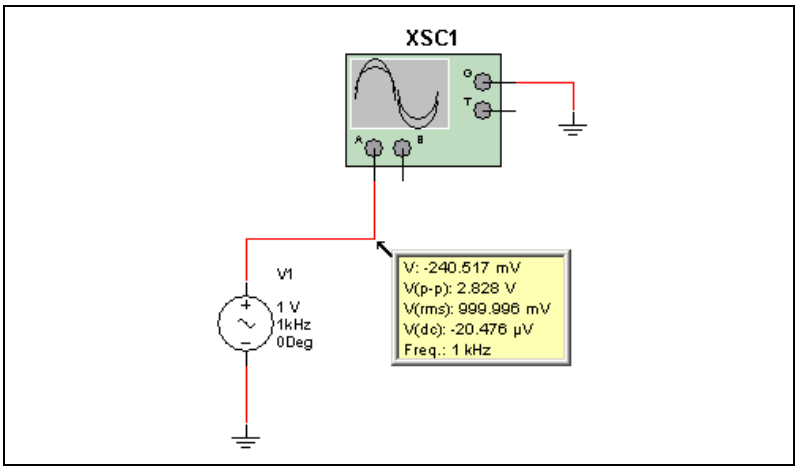
9. To hide a parameter (for example, **V(p-p)**), click beside the desired parameter in the **Show** column to toggle to **No**.
10. Use the **Minimum** and **Maximum** columns to set a parameter's range.
11. Optionally, change the number of significant digits for a displayed parameter in the **Precision** column.

Using the Dynamic Measurement Probe

Complete the following steps to use the **Measurement Probe** dynamically (that is, not placed at one point):

1. Select **Simulate»Run** or click the **Run/Resume Simulation** button to activate the circuit.

2. Click on the **Measurement Probe** button found in the **Instruments** toolbar. The probe is now attached to the mouse cursor.
3. Hover the probe on the point where you wish to take the measurement. The readings appear on the **Measurement Probe**, based on the dynamic probe settings as in the example shown below. Refer to the *Dynamic Probe Settings* section for more information.



Note When the circuit is simulating, and the probe is not placed on a wire or terminal, there are no readings on the probe.

4. To de-activate the probe, click on the **Measurement Probe** button or press <Esc>.

Note For accuracy of measurement, your circuit must have a ground attached.

Tip You can click to place the probe on any wire during simulation.

Using the Static Measurement Probe

Complete the following steps to connect a static (placed) **Measurement Probe** and take a reading:

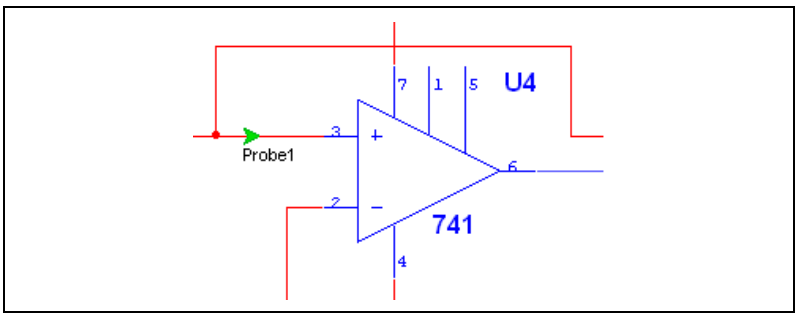
1. Click on the arrow in the **Measurement Probe** button found in the **Instruments** toolbar and select one of the following probe types from the list that appears:
   - **From dynamic probe settings**—The placed probe will use the settings entered using the **Simulate/Dynamic Probe Properties** command. Refer to the *Dynamic Probe Settings* section for more information.
   - **AC voltage**—The placed probe will measure V(p-p), V(rms), V(dc), and frequency.
   - **AC current**—The placed probe will measure I(p-p), I(rms), I(dc), and frequency.

- **Instantaneous voltage and current**—The placed probe will measure V and I.
- **Voltage with reference to probe**—Displays the **Reference Probe** dialog. Select the desired reference probe from the drop-down list (this is populated by a list of the probes already in the circuit). The placed probe will measure Vgain(dc), Vgain(ac), and Phase. If you select this probe type, a triangle appears beside the probe's RefDes.

2. Click to place the probe at the desired point on the circuit.
3. Select **Simulate»Run** or click on the **Run/Resume Simulation** button to activate the circuit.
4. The probe's window is populated with data.

Complete the following steps to hide the contents of the probe:

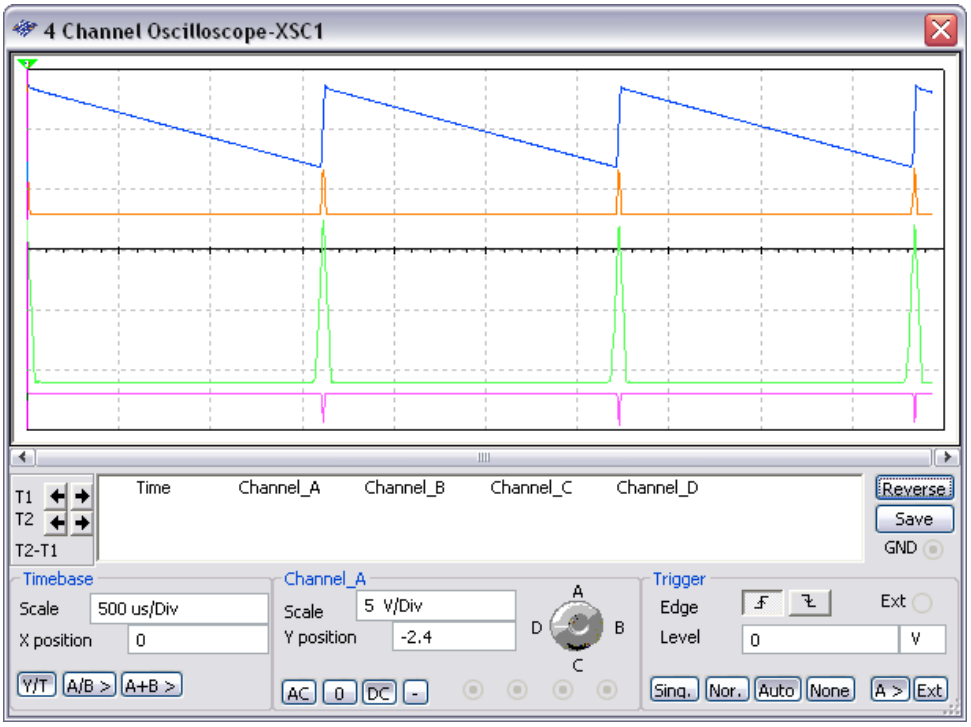
1. Right-click on the probe and de-select **Show Comment/Probe**. The placed probe now appears as an arrow, as shown in the example below.



To show the contents of probe, right-click on it and select **Show Comment/Probe**.

Four-Channel Oscilloscope

The four-channel oscilloscope allows the simultaneous monitoring of up to four different inputs.



To use the instrument, click on the **Four Channel Oscilloscope** button in the **Instruments** toolbar and click to place its icon on the workspace. The icon is used to wire the four channel oscilloscope to the circuit. Double-click on the icon to open the instrument face, which is used to enter settings and view measurements.

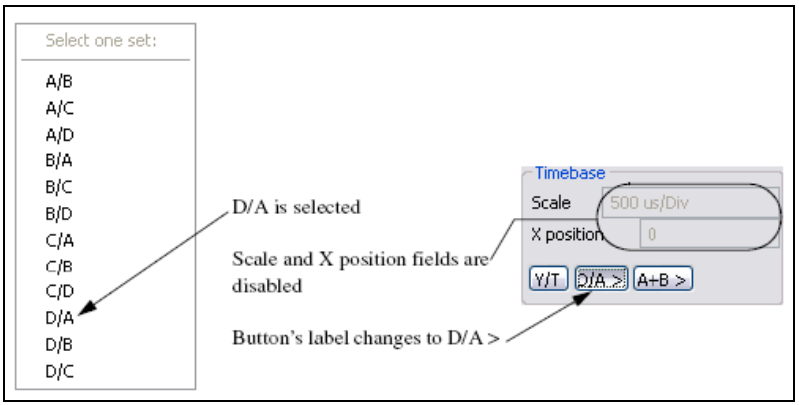
Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Four-channel Oscilloscope Settings

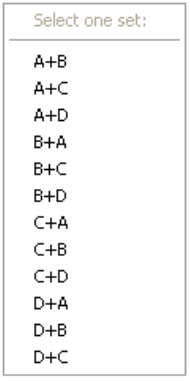
Either before, after or during a simulation, you can change the settings that are found at the bottom of the scope. In some cases, you may need to re-run the simulation to achieve accurate results.

Complete the following steps to change the settings:

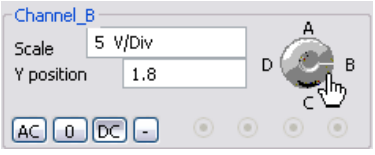
1. Adjust the settings in the **Timebase** box as follows:
   - **Scale** field—Click in this field to change the scale of the scope's horizontal (x-axis) when comparing magnitude against time (**Y/T**) or adding signals, for example, **A+B >** (see **Axes buttons** below).
   - **X position** field—Click in this field to change the signal's starting point on the x-axis. When set to 0, the signal starts at the left edge of the display. A positive value (for example, 2.00) shifts the starting point to the right, in this case two divisions. A negative value (for example, -3.00) shifts the starting point to the left, in this case three divisions.
   - **Axes buttons**—
     Y/T—Click to display waveform magnitude (y-axis) against time (x-axis).
     A/B >—Click to show input channel A against input channel B. (A is on the y-axis and B is on the x-axis). These settings display frequency and phase shifts, known as Lissajous patterns, or they can display a hysteresis loop. When this button is selected, the **Scale** and **X position** fields are disabled (greyed out).
     You can change the channels that are represented on the y-axis and x-axis by right-clicking on the **A/B >** button to display a pop-up menu. Remember that the first letter is the channel that is represented on the y-axis and the second letter is the channel that is represented on the x-axis. Once you have selected an item from the menu, the **A/B >** button's label changes to reflect your selection. For example, if you select **D/A >**, the button will read **D/A**.



A+B >—Click to add trace A to trace B.
You can change the channels that are added together by right-clicking on the **A+B >** button and making a selection from the pop-up menu that appears.



2. Adjust the settings in the **Channel_A** box as follows:
   - Channel dial—Click and drag on the dial to select the input channel that is being set in the **Scale** and **Y position** fields and input coupling buttons. Note that the name of the box changes as you move the dial.



3. **Scale** field—Click here to change the number of volts per division. This setting determines the scale of the y-axis. It also controls the x-axis scale when the **A/B >** button is selected in the **Timebase** box.

Note When running a simulation, the **Scale** selected in the displayed channel is the scale that is used on the graph. For example, if you have selected Channel B with the channel dial, and its **Scale** is set to 5 V/Div, all channels in the graph will be shown at 5 V/Div, even if channels A, C & D are set to 2 V/Div.

- **Y position** field—Click in this field to change the signal's starting point on the y-axis. When set to 0, the signal starts at the intersection with the x-axis. A positive value (for example, 2.00) shifts the starting point above the x-axis, in this case two divisions. A negative value (for example, -3.00) shifts the starting point below the x-axis, in this case three divisions. Using a different **Y position** setting for each trace (channel) helps distinguish individual waveforms.
- Input coupling buttons:

  AC button—Click to display only the AC component of a signal. This has the effect of placing a capacitor in series with the oscilloscope's probe. As on a real oscilloscope using AC coupling, the first cycle displayed is inaccurate. Once the signal's **DC** component has been calculated and eliminated in the first cycle, the waveforms are accurate.
  0 button—Click to display a flat reference line at the point of origin set in **Y position**.
  DC button—Click to display the sum of the AC and DC components of the signal.
  - button—Click to invert the selected channel's trace 180 degrees.

Note Do not place a coupling capacitor in series with an oscilloscope probe. The oscilloscope will not provide a path for current and the analysis will consider the capacitor improperly connected. Instead, choose **AC**.

4. Adjust the settings in the **Trigger** box as follows:
   - **Edge** buttons—Click the **Ascending Edge** button to start displaying the signal on its positive slope or rising signal. Click the **Descending Edge** button to start with the negative slope or falling signal.
   - **Level** fields—Enter the trigger level in the left field and the unit of measure in the right field. The trigger level is the point on the y-axis that must be crossed by the waveform before it is displayed.

A flat waveform will not cross the trigger level. To see a flat signal, make sure the triggering signal is set to **Auto**.

- Trigger buttons:
  Sing (single) button—Click to make the scope trigger one pass when the trigger level is met. Once the trace gets to the end of the scope face, it will not change until you click **Sing** again.
  Nor (normal) button—Click to make the scope refresh every time the trigger level is met.
  Auto (automatic) button—Click to display a flat signal, or if you wish to display signals as soon as possible. If **Auto** is selected, the **A >** and **Ext** buttons are disabled (greyed out).
  A > button—If selected, triggering is internal with reference to channel A. If you wish to change the internal reference channel, right-click on **A >** to display a pop-up menu and select the desired channel (A, B, C, or D). The label on the button changes to reflect your selection.

  None button—Click to have no trigger setting.

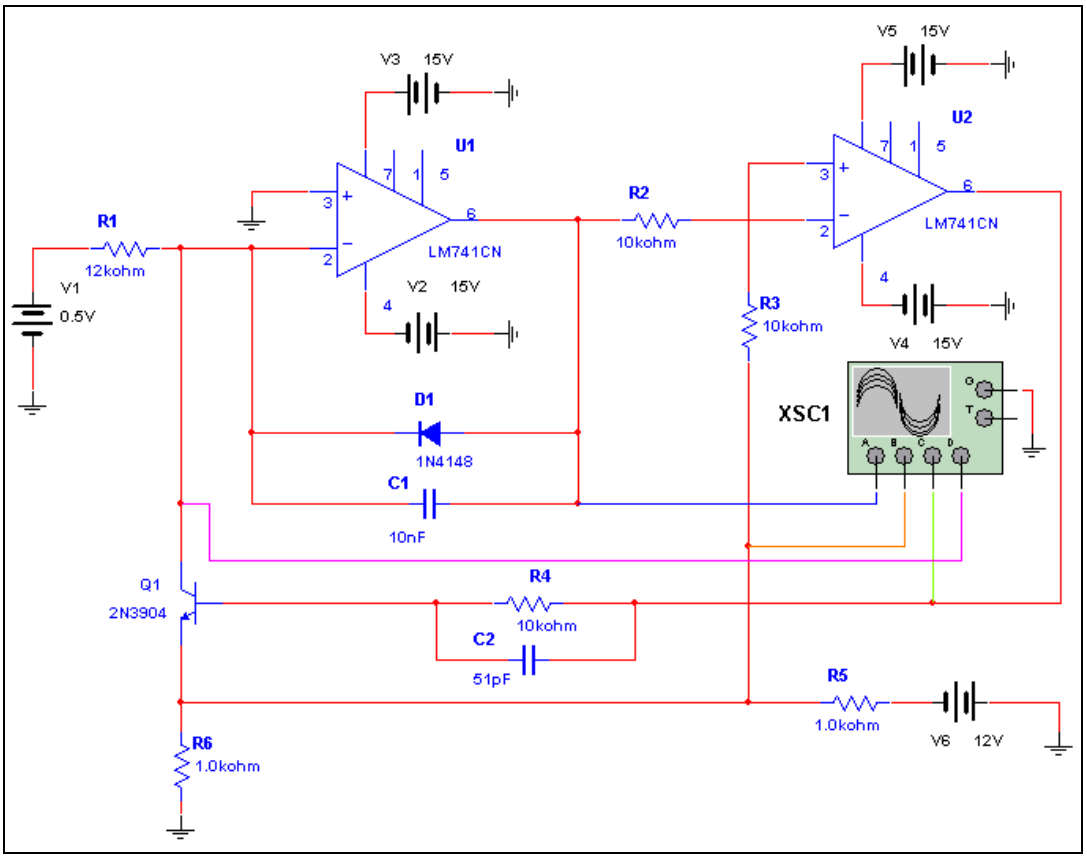
  Ext (external) button—Click if you wish to trigger the scope from an external source.

5. Optionally, click **Save** to save the results in one of the following file formats:
   - Scope data (`*.scp`)—Multisim oscilloscope format file.
   - LabVIEW measurement files (`*.lvm`)—Files created in National Instruments LabVIEW.
   - DIAdem files (`*.tdm`)—Files created in National Instruments DIAdem. Note that when you save data as a DIAdem file, two files are created; a header file (`.tdm`) and a binary file (`.tdx`).

Note If you choose either `.lvm` or `.tdm` files, the **Data resampling settings** dialog box displays. Along with the **Save** button found in the oscilloscope, you can also save simulation results from the **Grapher** window. Refer to the *Saving Files* section of Chapter 10, *Analyses*, for more information.

Viewing Four-channel Oscilloscope Results

This section uses the following circuit to demonstrate the use of the four-channel oscilloscope. This circuit is a sawtooth waveform generator with the scope attached to four different points.



Taking Readings

Complete the following steps to take readings with the four-channel oscilloscope:

1. Select **Simulate»Run** or click on the **Run/resume Simulation** button to activate the circuit.
2. Double-click on the four-channel oscilloscope's icon to open the instrument face.

Note You may open the instrument at any time before, during or after the simulation is run.

3. After a few seconds, either stop or pause the simulation. Optionally, click **Reverse** to show the display with a white background. The four traces on the oscilloscope's display represent the four input channels A

through D. The traces have the same colors that were selected for their wires in the Connecting the Four-channel Oscilloscope section.

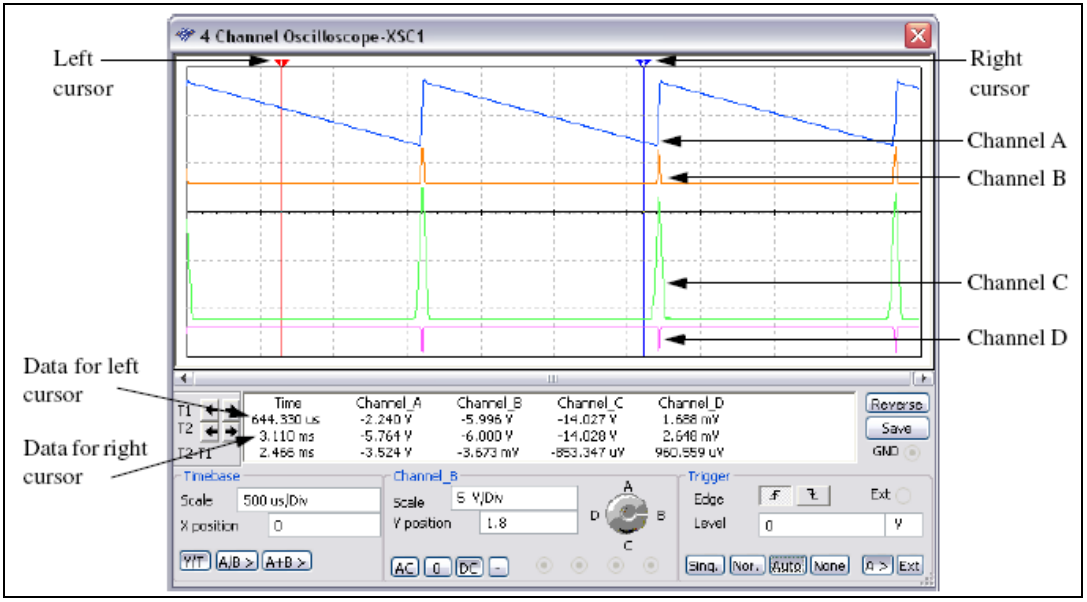
Note It is not necessary to stop or pause the simulation to view data or change the scope's settings.

Viewing Data

Either before, after or during a simulation, you can change the data viewing settings.

Complete the following steps to view data on the four-channel oscilloscope screen:

1. Place the mouse cursor over the left graph cursor (labeled "1") and drag the graph cursor to a point on the curves. Note that the data in the **T1** section of the scope change to reflect the points where the cursor crosses the curves. You may also drag the right graph cursor (labeled "2") to a desired point on the curves. When you do this the data in the **T2** row of the scope change.



2. You may also use the left and right arrows beside **T1** and **T2** to move the graph cursor.
3. You can also move either graph cursor to a specific x or y value on the selected curve by right-clicking on it to display a pop-up menu. Refer

to the *Cursor Pop-up Menu* section of Chapter 10, *Analyses*, for more information.

Connecting the Four-channel Oscilloscope

Complete the following steps to connect the four-channel oscilloscope:

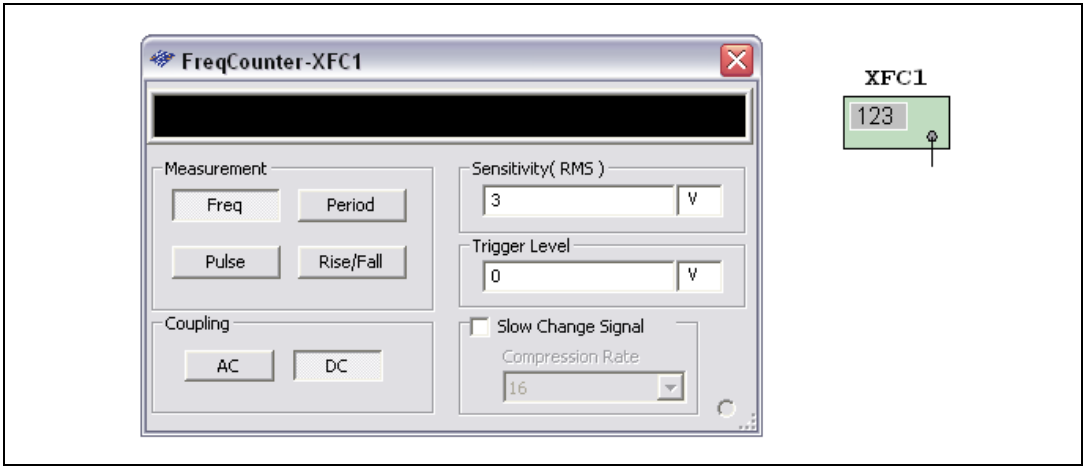
1. Click on the **4 Channel Oscilloscope** tool button and place the instrument icon in the desired location in your circuit.
2. Wire the icon to the desired points in your circuit using the following terminals as needed:
   - **A, B, C, D**—Input channels
   - **G**—Ground
   - **T**—External trigger
3. Select different colors for the wires leading to the four input channels to the oscilloscope. The traces of the four channels that appear on the oscilloscope will be represented by the colors chosen in this step.
   - Right-click on the wire leading to input channel A and select **Color Segment** from the pop-up that appears. The **Color** dialog box appears.
   - Click on the desired color for the wire and click **OK**.
   - Repeat for channels B – D.

Frequency Counter

The **Frequency Counter** is used to measure signal frequency.

To place the instrument, click on the **Frequency Counter** button in the **Instrument** toolbar and click to place its icon on the workspace. The icon is used to wire the **Frequency Counter** to the circuit.

Double-click on the icon to open the instrument face, which is used to enter settings and view measurements.



Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

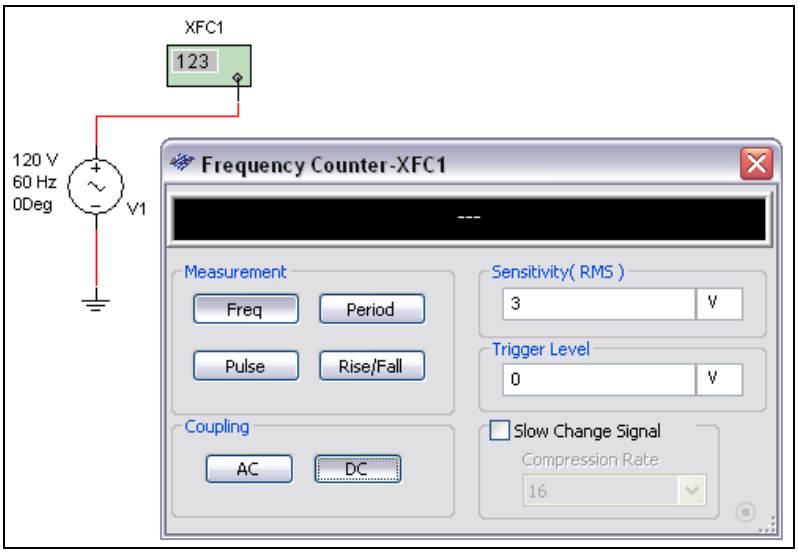
Using the Frequency Counter

Complete the following steps to measure a circuit using the frequency counter:

1. Click on the **Frequency Counter** tool button and place the icon in the desired location in your circuit. Wire the frequency counter to the point in the circuit that you wish to measure.

Note This example uses a simple AC source to provide a signal to the **Frequency Counter**.

2. Double-click on the icon to open the instrument face.



3. Set the instrument to read as desired, using the following controls:
   - **Measurement** box:
     Freq (frequency) button—Click to measure the frequency.
     Pulse button—click to measure the duration of a positive and negative pulse.
     Period button—Click to measure the duration of a single cycle.
     Rise/Fall button—Click to measure the rise and fall times of a single cycle.
   - **Coupling** box:
     AC button—Click to display only the AC component of a signal.
     DC button—Click to display the sum of the AC and DC components of the signal.
   - **Sensitivity (RMS)** box—Enter the sensitivity in the left field and the unit of measure in the right field.
   - **Trigger Level** box—Enter the trigger level in the left field and the unit of measure in the right field. The trigger level is the point that must be reached by the waveform before a reading is displayed.
   - **Slow Change Signal** checkbox—Enable when measuring frequencies below 20 Hz and change the **Compression Rate** as required to produce a flicker-free display.

Note The above settings may be changed before, during or after a simulation.

4. Select **Simulate»Run**. Since the **Freq** button is selected, the frequency at the point where the frequency counter is connected displays, as in the example shown below.

5. Other readings may be observed as follows:
   - **Pulse**—The duration of the positive and negative pulses display as shown below.

   - **Period**—The duration of a single cycle displays as shown below.

   - **Rise/Fall**—The rise and fall times of a single cycle display as shown below.

IV Analyzer

The **IV Analyzer** is used to measure the current-voltage curves of the following devices:

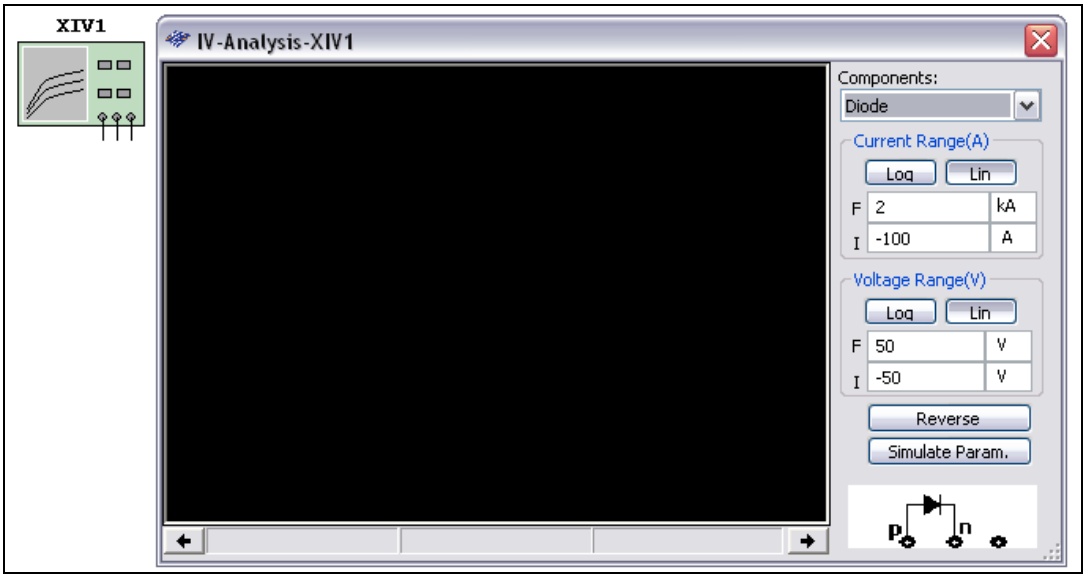
- Diode
- PNP BJT
- NPN BJT
- PMOS
- NMOS.

Note The **IV Analyzer** measures single components that are not connected in a circuit. You may measure devices that are already in circuits by disconnecting them first.

Using the IV Analyzer

Complete the following steps to use the **IV Analyzer** to measure a device's characteristics:

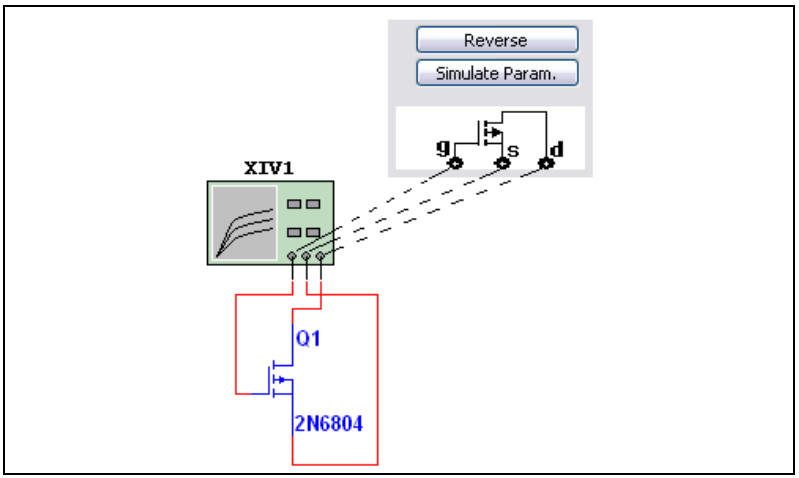
1. Click on the **IV Analysis** tool button, place its icon on the workspace and double-click on the icon to open the instrument.
2. Select the type of device that you are analyzing from the **Components** drop-down list, for example PMOS.



Note A PMOS FET is analyzed in this example.

3. Place the desired device on the workspace and wire it to the **IV Analyzer** following the symbol map that is shown on the instrument face.

Note The symbol changes depending on the device selected in the **Components** drop-down list.



Note If checking a device that is already in a circuit, disconnect it from the circuit and attach it to the **IV Analyzer** as shown above.

4. Click **Simulate Param.** to display the **Simulate Parameters** dialog box.

Note The contents of this dialog box change depending on the device selected in the **Components** drop-down list. Refer to the *Simulate Parameters Dialog Box* section for information about how this dialog box appears for other devices.

Optionally, change the default settings for V_{ds} (drain-source voltage) that appear in the following fields in the **Source Name: V_ds** box:

- **Start**—Enter the desired start V_{ds} for the sweep in the left section and the unit of measure in the right section
- **Stop**—Enter the desired stop V_{ds} for the sweep in the left section and the unit of measure in the right section
- **Increment**—Enter the desired size of the steps of V_{ds} for the sweep in the left section and the unit of measure in the right section. The points formed by these steps will be the points at which measurements will be calculated to form the curve(s) on the graph.

Optionally, change the default settings for V_{gs} (gate-source voltage) that appear in the following fields in the **Source Name: V_gs** box:

- **Start**—Enter the desired start V_{gs} for the sweep in the left section and the unit of measure in the right section

- **Stop**—Enter the desired stop V_{gs} for the sweep in the left section and the unit of measure in the right section
- **Num steps**—Enter the desired number of steps of V_{gs} for the sweep. There will be one curve for each value of V_{gs}.

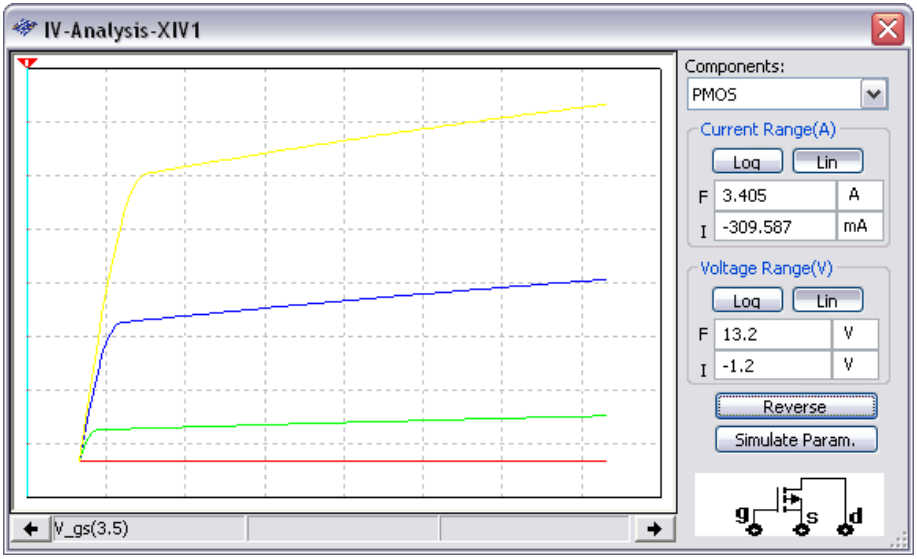
Normalize Data checkbox—Displays the V_{ds}values in the curves (x-axis) with positive values.

Click **OK** to save the settings and return to the main IV Analyzer dialog box.

5. Optionally, change the default scale buttons in the **Current Range(A)** and **Voltage Range(V)** boxes from **Lin** (linear) to **Log** (logarithmic). In this example, both settings were left at **Lin**.

Note There is no need to change the **F** (final) and **I** (initial) current and voltage settings. After the simulation is run, these fields are populated so that the curves fully display. Refer to the *Reviewing IV Analyzer Data* section for information about changing these fields to view specific areas of the curve.

6. Select **Simulate»Run**. The IV curves for the device are displayed. If desired, click on **Reverse** to change the display to a white background.



Note Refer to the *Reviewing IV Analyzer Data* section for more information.

7. Optionally, select **View»Grapher** to view the results of the simulation on the grapher.

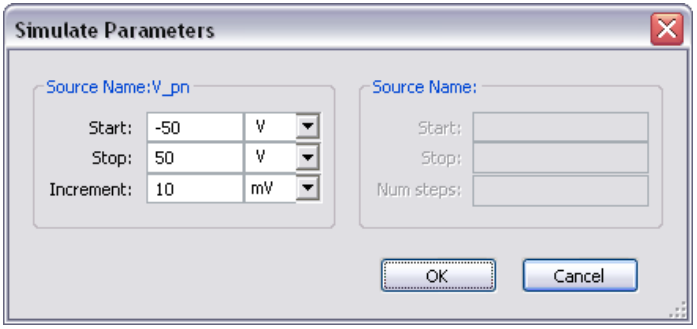
Simulate Parameters Dialog Box

The **Simulate Parameters** dialog box, which is described in step 4 above, changes depending on the device selected in the **Components** drop-down list. This section describes the data that appear in the dialog box when other component types are selected.

Diode Parameters

When **Diode** is selected in the **Components** drop-down list of the **IV Analyzer** dialog box and you click **Simulate Param.**, the **Simulate Parameters** dialog box that appears is populated with the following data:

Note The **Source Name** box is disabled for diode measurements.

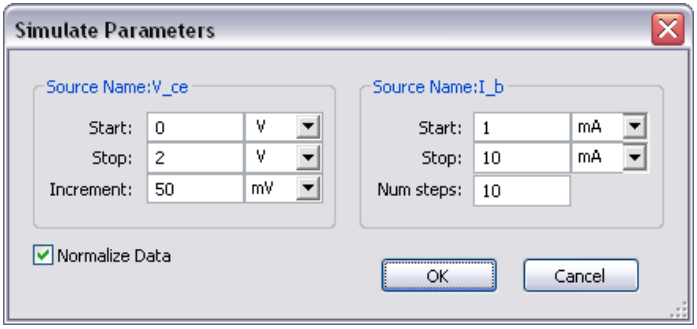


Optionally, change the default settings that appear in the following fields in the **SourceName: V_pn** box:

- **Start**—Enter the desired start V_{pn} for the sweep in the left section and the unit of measure in the right section.
- **Stop**—Enter the desired stop V_{pn} for the sweep in the left section and the unit of measure in the right section.
- **Increment**—Enter the desired size of the steps of V_{pn} for the sweep in the left section and the unit of measure in the right section. The points formed by these steps will be the points at which measurements will be calculated to form the curve on the graph.

PNP BJT Parameters

When **BJT PNP** is selected in the **Components** drop-down list of the **IV Analyzer** dialog box and you click **Simulate Param.**, the **Simulate Parameters** dialog box that appears is populated with the following data.



Optionally, change the default parameters for V_{ce} (collector-emitter voltage) that appear in the following fields in the **Source Name: V_ce** box:

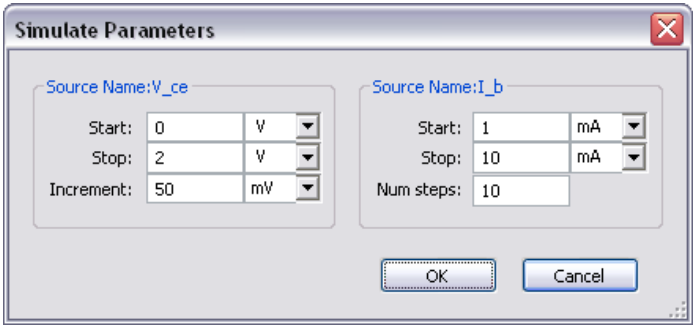
- **Start**—Enter the desired start V_{ce} for the sweep in the left section and the unit of measure in the right section.
- **Stop**—Enter the desired stop V_{ce} for the sweep in the left section and the unit of measure in the right section.
- **Increment**—Enter the desired size of the steps of V_{ce} for the sweep in the left section and the unit of measure in the right section. The points formed by these steps will be the points at which measurements will be calculated to form the curve(s) on the graph.

Optionally, change the default parameters for I_b (base current) that appear in the following fields in the **Source Name: I_b** box:

- **Start**—Enter the desired start I_b for the sweep in the left section and the unit of measure in the right section.
- **Stop**—Enter the desired stop I_b for the sweep in the left section and the unit of measure in the right section.
- **Num steps**—Enter the desired number of steps of I_b. There will be one curve for each value of I_b.
- **Normalize Data** checkbox—Displays the V_{ce} values on the curves (x-axis) with positive values.

NPN BJT Parameters

When **BJT NPN** is selected in the **Components** drop-down list of the **IV Analyzer** dialog box and you click **Simulate Param.**, the **Simulate Parameters** dialog box that appears is populated with the following data.



Optionally, change the default settings parameters for V_{ce} (collector-emitter voltage) that appear in the following fields in the **Source Name: V_ce** box:

- **Start**—Enter the desired start V_{ce} for the sweep in the left section and the unit of measure in the right section.
- **Stop**—Enter the desired stop V_{ce} for the sweep in the left section and the unit of measure in the right section.
- **Increment**—Enter the desired size of the steps of V_{ce} for the sweep in the left section and the unit of measure in the right section. The points formed by these steps will be the points at which measurements will be calculated to form the curve(s) on the graph.

Optionally, change the default parameters for I_b (base current) that appear in the following fields in the **Source Name: I_b** box:

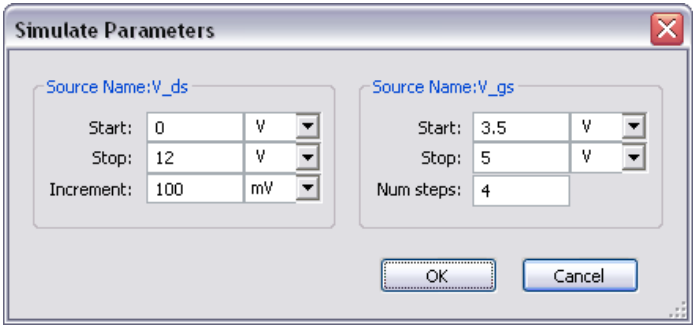
- **Start**—Enter the desired start I_b for the sweep in the left section and the unit of measure in the right section.
- **Stop**—Enter the desired stop I_b for the sweep in the left section and the unit of measure in the right section.
- **Num steps**—Enter the desired number of steps for I_b. There will be one curve for each value of I_b.

PMOS FET Parameters

See step 4 for details on the **Simulate Parameter** dialog box when used with a **PMOS** FET.

NMOS FET Parameters

When **NMOS** is selected in the **Components** drop-down list of the **IV Analyzer** dialog box and you click **Simulate Param.**, the **Simulate Parameters** dialog box that appears is populated with the following data.



Optionally, change the default parameters for V_{ds} (drain-source voltage) that appear in the following fields in the **Source Name: V_ds** box:

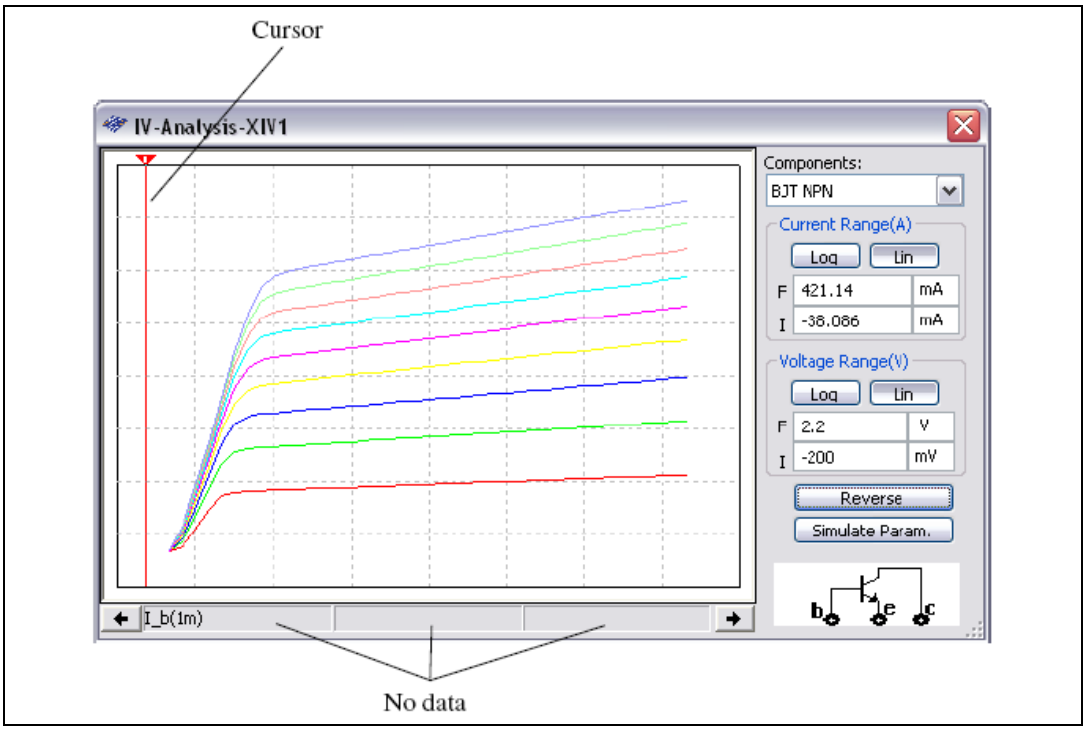
- **Start**—Enter the desired start V_{ds} for the sweep in the left section and the unit of measure in the right section.
- **Stop**—Enter the desired stop V_{ds} for the sweep in the left section and the unit of measure in the right section.
- **Increment**—Enter the desired size of the steps of V_{ds} for the sweep in the left section and the unit of measure in the right section. The points formed by these steps will be the points at which measurements will be calculated to form the curve(s) on the graph.

Optionally, change the default parameters for V_{gs} (gate-source voltage) that appear in the following fields in the **Source Name: V_gs** box:

- **Start**—Enter the desired start V_{gs} for the sweep in the left section and the unit of measure in the right section.
- **Stop**—Enter the desired stop V_{gs} for the sweep in the left section and the unit of measure in the right section.
- **Num steps**—Enter the desired number of steps of V_{gs} for the sweep. The points formed by these steps will be the points at which measurements will be calculated to form the curve(s) on the graph.

Reviewing IV Analyzer Data

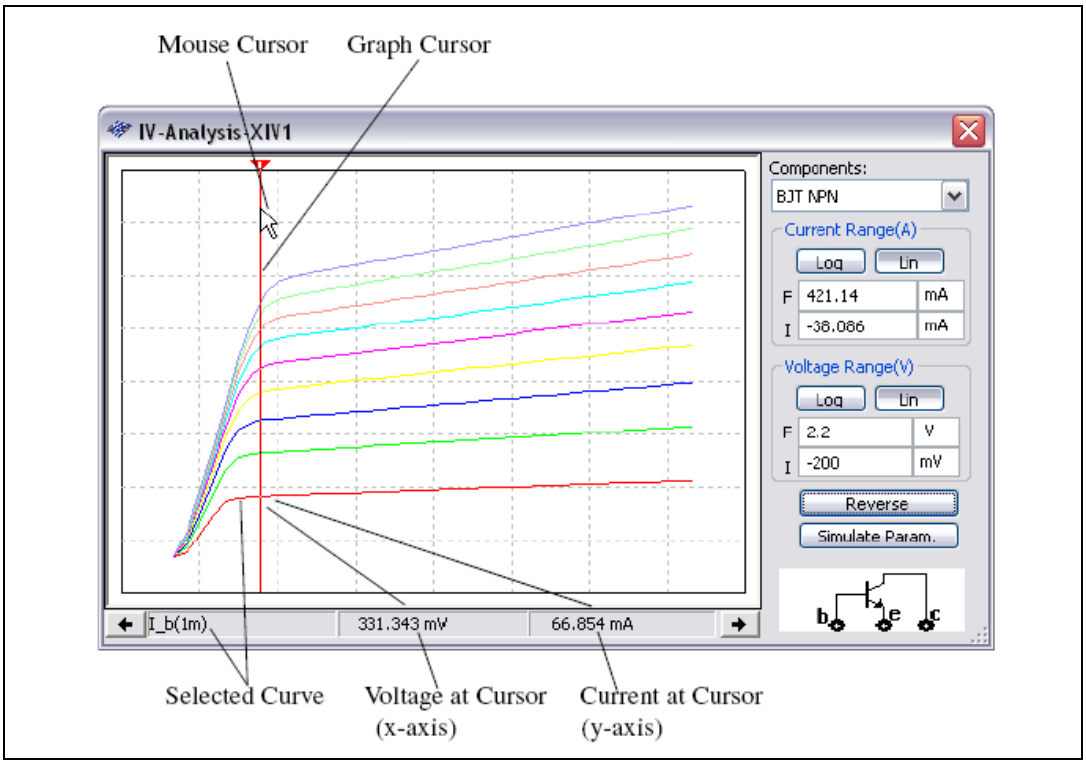
After you run an IV analysis, you will have a graph similar to the one in the figure below. Note that the cursor is not on any of the curves and the three fields at the bottom of the graph are empty.



Note This section uses data from the analysis of an NPN BJT.

Complete the following steps to view data in the **IV Analyzer** dialog box:

1. As shown in the example below, place the mouse cursor over the graph cursor and drag the graph cursor to a point on the curves. The three fields at the bottom of the graph now contain data which corresponds to a base current (I_b) of 1 mA. Ib(1m) is represented by the curve at the bottom of the graph.



2. To select another curve, for example, I_b = 2 mA, click on that curve with your mouse cursor. The fields at the bottom of the graph now contain the data from the point where the graph cursor crosses the selected curve.
3. You may also use the left and right that appear below the displayed graph arrows to move the graph cursor.
4. You can also move the graph cursor to a specific x or y value on the selected curve by right-clicking on it to display a pop-up menu. Refer to the *Cursor Pop-up Menu* section of Chapter 10, *Analyses*, for more information.
5. To view specific areas of the graph, change the data in the **Current Range(A)** and **Voltage Range(V)** boxes:
   - **Current Range(A)** (y-axis)
     I field—Enter initial current to be displayed on graph in left field and unit of measure in right field.
     F field—Enter final current to be displayed on graph in left field and unit of measure in right field.

- **Voltage Range(V)** (x-axis)
  I field—Enter initial voltage to be displayed on graph in left field and unit of measure in right field.
  F field—Enter final voltage to be displayed on graph in left field and unit of measure in right field.

Agilent Simulated Instruments

The following virtual simulated Agilent instruments are included in Multisim:

- Function Generator 33120A
- Multimeter 34401A
- Oscilloscope 54622D

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Agilent Simulated Function Generator

The Agilent Technologies 33120A is a high-performance 15 MHz synthesized function generator with built-in arbitrary waveform capability. Refer to the Agilent website for a PDF copy of this instrument's user guide.

Supported Features

Most of the features that are documented in the Agilent 33120A User's Guide are available in its virtual simulated version. These include:

- **Standard Waveforms**—Sine, Square, Triangle, Ramp, Noise, DC volts.
- **System Arbitrary Waveforms**—Sinc, Negative Ramp, Exponential Rise, Exponential Fall, Cardiac.
- **User Defined Arbitrary Waveforms**—Any type of waveform with 8 – 256 points.
- **Modulations**—NON, AM, FM, Burst, FSK, Sweep.
- **Memory Sections**—Four memory sections named #0 – #3. #0 is the system default.
- **Trigger Modes**—Auto/Single for Burst and Sweep modulation only.
- **Display Digital**—4 – 8.
- **Display Voltage**—Three modes: Vpp, Vrams and dBm.

- **Edit Digital Value**—Change the displayed value with cursor buttons or number keys, or use the knob, or use the ENTER NUMBER key to key in the number directly.
- **Menu Operation**:
  - A. MODulation MENU—1: AM SHAPE, 2:FM SHAPE, 3:BURST CNT, 4:BURST RATE, 5:BURST PHAS, 6:FSK FREQ, 7:FSK RATE.
  - B. SWP MENU—1:START F, 2:STOP F, 3:SWP TIME, 4:SWP MODE.
  - C. EDIT MENU—1:NEW ARB, 2:POINTS, 3:LINE EDIT, 4:POINT EDIT, 5:INVERT, 6:SAVE AS, 7:DELETE.
  - D. SYStem MENU—1:COMMA.

Features Not Supported

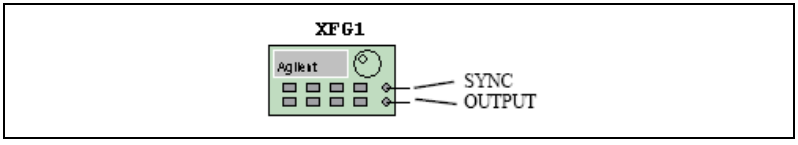
The following features are not supported in the simulated version of the function generator included in Multisim:

- Remote mode.
- Terminals on rear panel.
- Self-test.
- Hardware error detect.
- Calibration.

Using the Agilent Function Generator

Complete the following steps to connect the Agilent 33120A Function Generator to a circuit:

1. Click on the **Agilent Function Generator** tool button, place its icon on the workspace and double-click on the icon to open the instrument. Click the power button to switch on the instrument.
2. Wire the icon to the circuit following the pin key below.



3. Refer to the PDF user guide for complete instructions on the use of this device.

Agilent Simulated Multimeter

The simulated Agilent Technologies 34401A Multimeter is a 6 1/2-digit, high-performance digital multimeter. Refer to the Agilent website for a PDF copy of this instrument's user guide.

Supported Features

Most of the features that are documented in the Agilent 34401A User's Guide are available in its virtual version. These include:

- **Measure Modes**—Voltage DC/AC, Current DC/AC, Resistor in two wires, Frequency of input signal's voltage wave, Period of input signal's voltage wave, Continuity test, Diode test, Ratio test.
- **Functions**—Null (relative measurement), Min-Max (store minimum and maximum readings), dB (display on voltage value), dBm (display on voltage value), Limit Test (test readings with a lower threshold value and a high threshold value).
- **Trigger Modes**—Auto/Manual.
- **Display Modes**—Auto/Manual.
- **Display Digital**—4 1/2 – 6 1/2.
- **Reading Hold**—Yes.
- **Reading Memory**—Yes.
- **Edit Digital Value**—Change the displayed value with cursor buttons or number keys.

All other features and operations are based on the Agilent 34401A User's Guide.

Features Not Supported

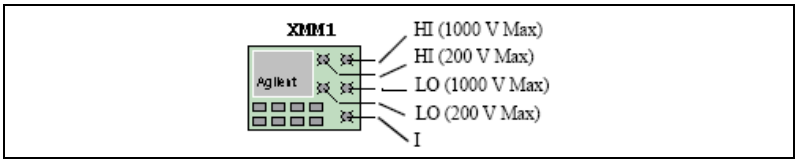
The following features are not supported in the simulated version of the **Multimeter** included in Multisim:

- Remote mode.
- Commands mode.
- Terminals on rear panel.
- Self-test.
- Hardware error detect.
- Calibration.

Using the Agilent Multimeter

Complete the following steps to connect the simulated Agilent 34401A Multimeter to a circuit:

1. Click on the **Agilent Multimeter** tool button, place its icon on the workspace and double-click on the icon to open the instrument. Click the power button to switch on the instrument.
2. Wire the icon to the circuit following the pin key below.



3. Refer to the PDF user guide, available on the Agilent website, for complete instructions on the use of this device.

Agilent Simulated Oscilloscope

The simulated Agilent Technologies 54622D Oscilloscope is a 2-channel +16 logic channels, 100-MHz bandwidth oscilloscope. Refer to the Agilent website for a PDF copy of this instrument's user guide.

Supported Features

Most of the features that are documented in the Agilent 54622D User's Guide are available on its simulated version. These include:

- **Running Mode**—Auto, Single, Stop.
- **Trigger Mode**—Auto, normal, auto-level.
- **Trigger Type**—Edge trigger, pulse trigger, pattern trigger.
- **Trigger Source**—Analog signals, digital signals, external trigger signal.
- **Display Mode**—Main, delay, roll, XY.
- **Signal Channels**—2 analog channels, 1 math channel, 16 digital channels, 1 probe signal for testing purposes.
- **Cursors**—4 cursors.
- **Math Channel**—FFT, multiply, subtract, differentiate, integrate.
- **Measurements**—Cursor information, sampling information, frequency, period, peak-peak, maximum, minimum, rise time, fall time, duty cycle, RMS, width, average, X at max.

- **Display Controls**—Vector/point on traces, trace width, background color, board color, grid color, cursor color.
- **Auto-scale/Undo**—Yes.
- **Print Traces**—Yes.
- **File**—Save data into a DAT format file; can be converted and displayed on the system graph window.
- **Soft-button Menu Operation**:
  - Main Display MENU—1:Main, 2: Delayed, 3:Roll, 4:XY, 5:Vernier, 6:Time Ref.
  - Cursor MENU—1:Source, 2:X Y, 3:X1, 4:X2, 5:X1 X2, 6:Cursor.
  - Quick Measure MENU—1:Source, 2:Clear Measure, 3:Frequency, 4:Period, 5:Peak_Peak, 6:Maximum, 7:Minimum, 8:Rise Time, 9:Fall Time, 10:Duty Cycle, 11:RMS, 12:+Width, 13:-Width, 14:Average, 15:X at Maximum.
  - Acquire MENU—1:Normal, 2:Average, 3:Args.
  - Display MENU—1:Clear, 2:Grid, 3:Background, 4:Boarder, 5:Vector.
  - Auto-Scale MENU—1:Undo Auto-scale.
  - Save MENU—1:Save.
  - Quick Print MENU—1:Print.
  - Utility MENU—1:Sampling Information, 2:Default Settings.
  - Edge MENU—1:Source, 2:Shape.
  - Pulse MENU—1:Source, 2:Shape, 3:Operation, 4:Less Value, 5:Great Value.
  - Pattern MENU—1:Source, 2:L, 3:H, 4:X, 5:Up Edge, 6:Down Edge.
  - Mode Coupling MENU—1:Mode, 2:Hold-off Value.
  - Analog Channel MENU—1:Coupling, 2:Vernier, 3:Invert.
  - Math Channel MENU—1:Setting, 2:FFT, 3:Multiply, 4:Subtract, 5:Differentiate, 6:Integrate.
  - Math FFT MENU—1:Source, 2:Span, 3:Center, 4:Scale, 5:Offset.
  - Math 1*2/1-2 MENU—1:Scale, 2:Offset.
  - Math Diff/Inte MENU—1:Source, 2:Scale, 3:Offset.
  - Digital Channel MENU—1:Select Channel/Enable/Disable, 2:Shape, 3:Threshold, 4:User Value.

- **Intensity**—Adjusts the intensity of the traces. If you print the output of the scope with the trace intensity as the lowest setting, each trace on the printout will have a diffent line style. If you increase the intensity, all the trace lines in the printout will be solid.

Features Not Supported

The following features are not supported in the simulated version of the oscilloscope included in Multisim:

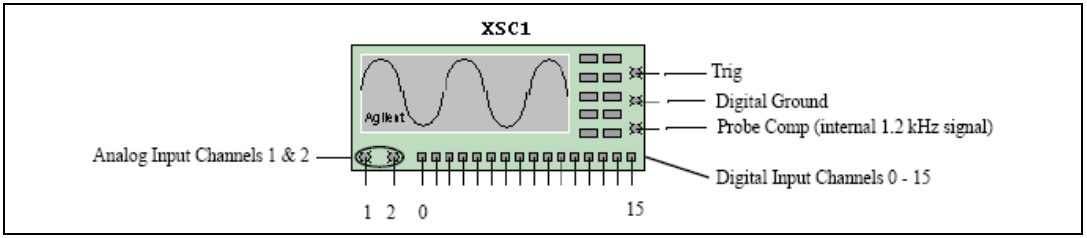
- Remote mode.
- Terminals on rear panel.
- Self-test.
- Hardware error detect.
- Calibration.
- Language selection.
- Infinite Persistence Operation mode.
- Label button for editing on digital channel's label.
- Delay and phase measurement.
- Overshoot and preshoot measurement.
- Clock setting.
- Cursor has normal mode only.
- Peak detection and real time data acquirement.
- Noise reject and high frequency reject modes in data coupling.
- Duration trigger, IC trigger, Sequence trigger.
- Bandwidth limit feature.
- Saving data as `.bmp`, `.tiff` or `.csv` files.

Using the Agilent Oscilloscope

Complete the following steps to connect the simulated Agilent 54622D Oscilloscope to a circuit:

1. Click on the **Agilent Oscilloscope** tool button, place its icon on the workspace and double-click on the icon to open the instrument. Click on power button to switch on the instrument.

2. Wire the icon to the circuit following the pin key below:



3. Refer to the PDF copy of the instrument's user guide for complete instructions on the use of this device.

Tip For optimum simulation speed, set the time scale on the scope near to the value you would like to display during simulation.

Tektronix Simulated Instruments

The simulated Tektronix TDS 2024 oscilloscope is included in Multisim.

Note If you are not familiar with connecting and adjusting the settings of the instruments, refer to the *Adding an Instrument to a Circuit* and *Using the Instrument* sections before using this instrument.

Tektronix Simulated Oscilloscope

The simulated Tektronix TDS 2024 is a 4-channel, 200-MHz oscilloscope. Refer to the Tektronix website for a PDF copy of this instrument's user guide.

Supported Features

Most of the features that are documented in the Tektronix TDS 2024's User Guide are available in its simulated version. These include:

- **Running Mode**—Auto; Single; Stop.
- **Trigger Mode**—Auto; Normal.
- **Trigger Type**—Edge Trigger; Pulse Trigger.
- **Trigger Source**—Analog signals; Extent trigger signal.
- **Display Mode**—Main; Window; XY, FFT, Trig View.
- **Signal Channels**—4 analog channels; 1 math channel; one probe signal in 1kHz for testing purposes.

- **Cursors**—4 cursors.
- **Math Channel**—FFT; +; –.
- **Measurements**—Cursor information; Frequency; Period; Peak-Peak; Maximum; Minimum; Rise time; Fall time; RMS; Mean.
- **Display Controls**—Vector/Point on traces; Contrast color control.
- **Auto-setup**—Yes.
- **Print Traces**—Yes.

Control Buttons Operation

- **Run/Stop Button**—Start or Stop sampling in multiple-triggers.
- **Single Seq. Button**—Start sampling in one-trigger.
- **Trig View Button**—See current trigger signal wave and trigger level.
- **Force Trig Button**—Immediately to start a trigger.
- **Set to 50% Button**—Move trigger level to the mean value of the trigger signal.
- **Set to Zero Button**—Set time offset position to zero.
- **Help Button**—Goes to instrument help topic.
- **Print Button**—Print the graph to printer.
- **Soft Menu Buttons**—To support the menu controls.

Soft Button Menu Operation

The soft button menu is a subset of the Tektronix Oscilloscope TDS 2024's:

- **Save/Recall MENU**—1: Setup; 2: Save; 3: Recall.
- **Measure MENU**—Five measure areas, each one includes second a level menu to set a source channel and measure type under: Frequency, Period, Peak_peak, Maximum, Minimum, Rise time, Fall time; RMS, Mean.
- **Acquire MENU**—1: Sample, 2: Average, 3: Average value.
- **Auto Set MENU**—Shows one of three menu lists based on the signal curve type. A) (SIN Curve) 1: Multiple; 2: Single; 3: FFT; 4: Undo, (Pulse curve) 1: Multiple; 2: Single; 3: Slope Up; 4: Slope Down; 5: Undo, (Unknow curve) 1: Mean Value; 2: Peak-Peak Value.
- **Utility MENU**—1: System status, it includes second level menu: 1: Horizontal status; 2: Vertical CH1-CH2 status; 3: Vertical CH3-CH4 status; 4: Trigger status; 5: Misc. status.

- **Cursor MENU**—1: Type; 2: Source.
- **Display MENU**—1: Type, 2: Format, 3: Contrast Increase, 4: Contrast Decrease.
- **Default Setup MENU**—1: Undo Default Setup.
- **Trig MENU**—It will show one of three menu lists: A) (Edge Trigger) 1: Type; 2: Source; 3: Slope; 4: Mode; 5: Coupling; B) (Pulse Trigger page 1) 1: Type; 2: Source; 3: When; 4: Pulse Width; 5: More Page; C) (Pulse Trigger page 2) 1: Type; 2: Polarity; 3: Mode; 4: Coupling; 5) More Page.
- **Channel MENU**—1: Coupling, 2: Volts/Div, 3: Invert.
- **Math channel MENU**—Shows one of three sub menu lists: A) (+) 1: Operation; 2: CH1+CH2; 3: CH3+CH4; B) (-) 1: Operation; 2: CH1-CH2; 2: CH2-CH1; 3: CH3-CH4; 4: CH4-CH3; C) (FFT) 1: Operation; 2: Source; 3: Window.
- **Horizontal MENU**—1: Main; 2:Window Zone; 3: Window; 4: Trig Knob selection.

Features Not Supported

The following features are not supported in the simulated version of the oscilloscope:

- Remote mode (without external device and RS 232 interface)
- Terminals on rear panel
- Probe check
- Scan mode display
- Language selection, Error Log and Do Self Calibration on menu item UTILITY
- Saving Wave form, Ref A – Ref D on menu item SAVE/RECALL
- Video Trigger on menu item TRIG MENU
- Trigger coupling of Noise Reject, HF Reject and LF Reject on menu item TRIG MENU
- Cursor source of Ref A – Ref D on menu item CURSOR
- Persistence display mode on menu item DISPLAY
- Peak detect acquire on menu item ACQUIRE
- FFT Zoom on menu item MATH
- BW limit and Probe selection on menu item CHANNEL.

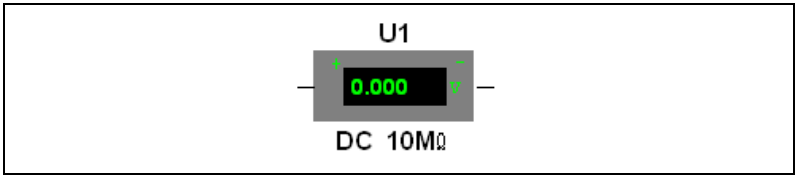
Using the Tektronix Oscilloscope

Complete the following steps to connect the simulated Tektronix TDS 2024 Oscilloscope to a circuit:

1. Click on the **Tektronix Oscilloscope** tool button, place its icon on the workspace and double-click on the icon to open the instrument. Click on the Power button to switch on the instrument.
2. Wire the icon to the circuit following the pin key below:
   - **1, 2, 3, 4**—analog input channels 1 – 4
   - **T**—trigger
   - **P**—probe comp (internal 1 kHz signal)
   - **G**—ground
3. Refer to the PDF copy of the instrument's user guide for complete instructions on the use of this device.

Voltmeter

The voltmeter offers advantages over the multimeter for measuring voltage in a circuit. It takes up less space in a circuit and you can rotate its terminals to suit your layout.



Using the Voltmeter

Resistance (1.0 Ω - 999.99 TW)

The voltmeter is preset to a very high resistance which generally has no effect on a circuit. If you are testing a circuit that itself has very high resistance, you may want to increase the voltmeter's resistance to get a more accurate reading. (However, using a voltmeter with very high resistance in a low-resistance circuit may result in a mathematical round-off error.)

To change the resistance of the voltmeter, double-click on it and change the value in the **Resistance** field that is found in the **Value** tab of the property dialog that displays.

Mode (DC or AC)

The voltmeter can measure DC or AC voltage. In DC mode, any AC component of the signal is eliminated so that only the DC component of the signal is measured. In AC mode, any DC component is eliminated so that only the AC component is measured. When set to AC, the voltmeter displays the root-mean-square (RMS) value of the signal.

To change the voltmeter's mode, double-click on it and select either **DC** or **AC** in the **Mode** field that is found in the **Value** tab of the property dialog that displays.

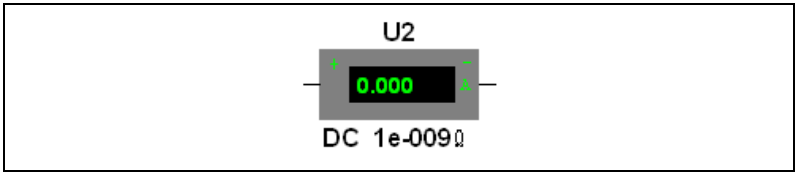
Connecting a Voltmeter

Connect the voltmeter in parallel with the load, attaching the probes to connectors on either side of the load you want to measure. When a circuit is activated and its behavior is simulated, the voltmeter displays the voltage across the test points. (The voltmeter may also display interim voltages before the final steady-state voltage is reached.)

Note The voltmeter is found in the **Indicator** component group. It is not found on the **Component** toolbar. If a voltmeter is moved after the circuit has been simulated, activate the circuit again to get a reading.

Ammeter

The ammeter offers advantages over the multimeter for measuring current in a circuit. It takes up less space in a circuit and you can rotate its terminals to suit your layout.



Using the Ammeter

Resistance (1.0 pΩ - 999.99 W)

The ammeter's resistance is preset to a low resistance. If you are testing a circuit that has low resistance, you can lower the ammeter's resistance even further to get a more precise measurement. (However, using an ammeter with very low resistance in a high-resistance circuit may result in a mathematical round-off error.)

To change the resistance of the ammeter, double-click on it and change the value in the **Resistance** field that is found in the **Value** tab of the property dialog that displays.

Mode (DC or AC)

The ammeter is preset to DC mode, which measures only the DC component of a signal. If you want to measure the current from an AC source, change the mode to AC. When set to AC, the ammeter displays the root-mean-square (RMS) value of the alternating signal.

To change the ammeter's mode, double-click on it and select either **DC** or **AC** in the **Mode** field that is found in the **Value** tab of the property dialog that displays.

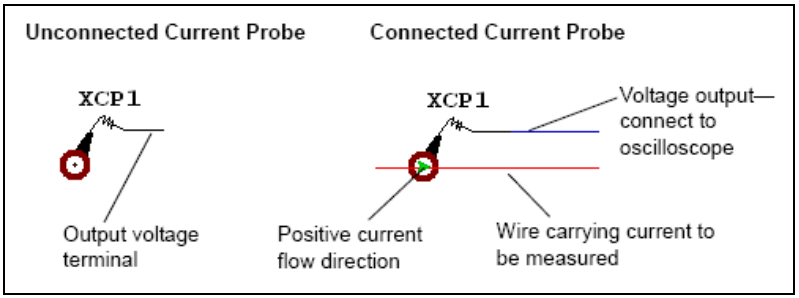
Connecting an Ammeter

Like a real ammeter, the simulated ammeter must be connected in series at nodes you want to measure. The negative terminal is on the side with the heavy border. If an ammeter is moved after the circuit has been simulated, activate the circuit again to get a reading.

Note The voltmeter is found in the **Indicator** component group. It is not found on the **Component** toolbar.

Current Probe

The **Current Probe** emulates the behavior of industrial clamp-on current probes, which convert the current flowing in a wire to a voltage at the device's output terminal. The output terminal is then connected to an oscilloscope, where the current is determined based on the voltage to current ratio of the probe.



Complete the following steps to place a **Current Probe** on your schematic:

1. Click on the **Current Probe** button found in the **Instruments** toolbar. The **Current Probe** is now attached to the cursor.
2. Drag the image of the **Current Probe** to the desired wire and click to place it. (You cannot place the **Current Probe** on a junction).
3. Place an oscilloscope on the workspace and connect the output terminal of the **Current Probe** to the oscilloscope.

Current Probe Properties

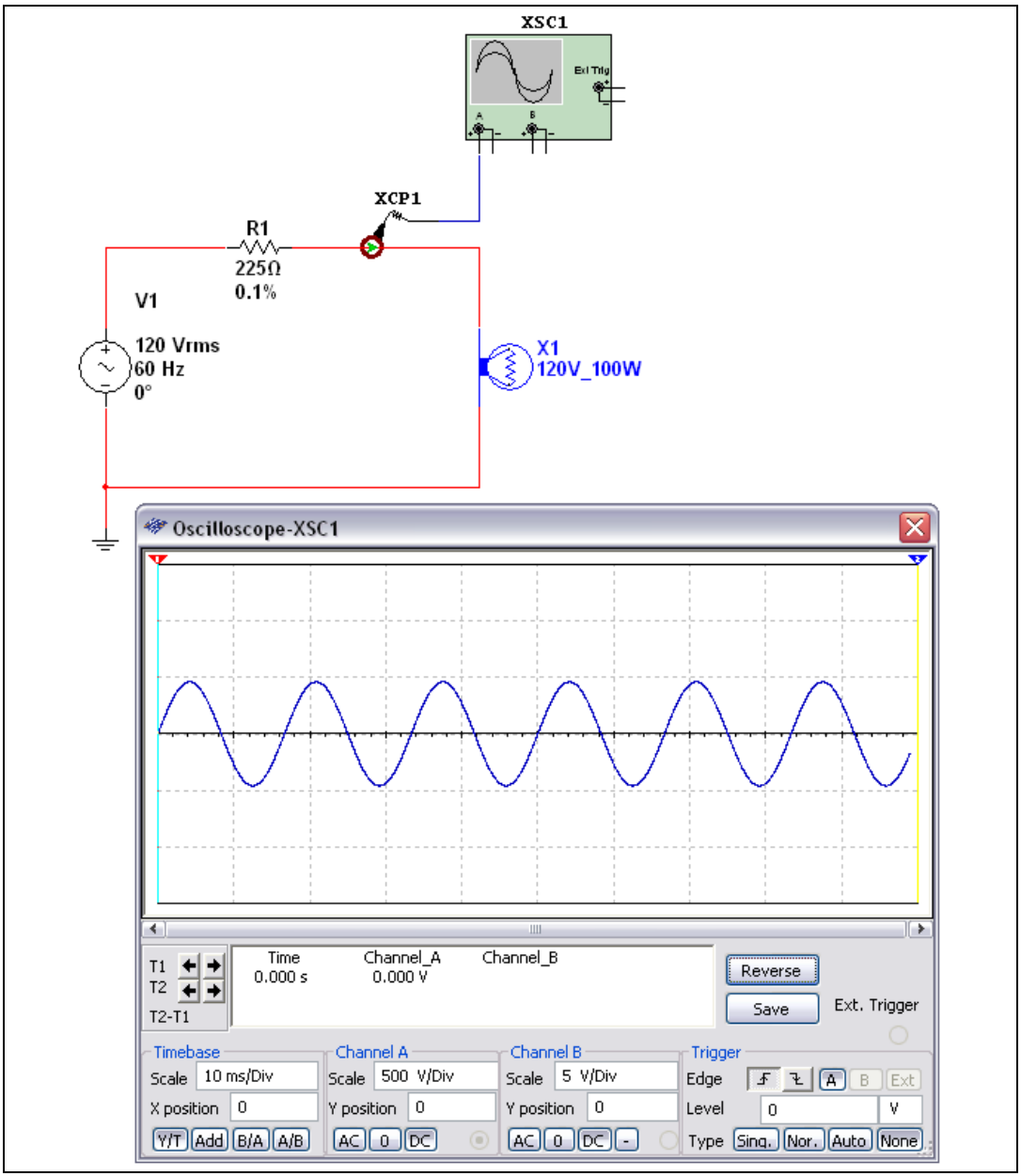
To emaluate the behavior of a typical real-world current probe, the default ratio of the current probe's output voltage to the current being measured is 1 V/mA.

Complete the following steps to change the ratio:

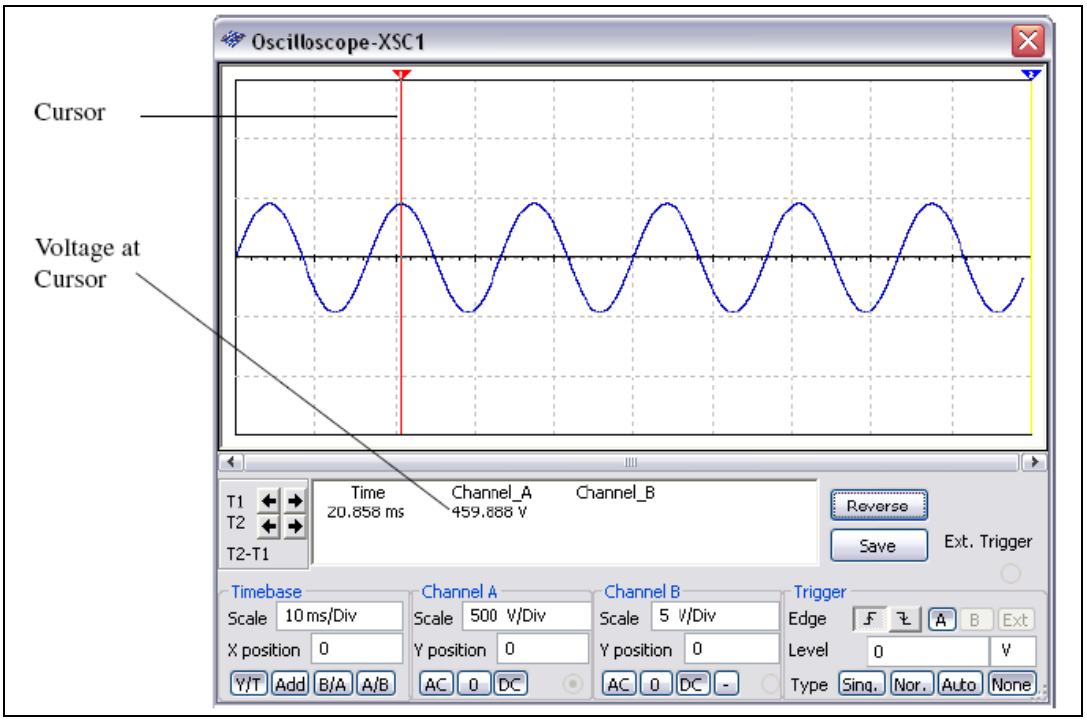
1. Double-click on the **Current Probe** to display the **Current Probe Properties** dialog box.
2. Change the value in the **Ratio of Voltage to Current** field and click **Accept**.

Complete the following steps to measure the current:

1. Double-click on the oscilloscope icon to show its instrument face.
2. Simulate the circuit and observe the output on the oscilloscope.
3. Stop or pause the simulation when a trace displays on the oscilloscope. It may be necessary to adjust the scale setting to display the trace in a meaningful way.



4. Drag one of the oscilloscope's cursors to a point on the trace and note the voltage reading.



Note Refer to the Oscilloscope section for complete information about the oscillope.

5. Convert the voltage to current using the ratio that is set for the probe. For details, see the *Current Probe Properties* section.

   In the example shown above, the voltage reading on the oscilloscope is 459.908 V. Since the ratio for this example was left at the default of 1 V/mA, this corresponds to a current of 459.908 mA.

Tip For a direct 1:1 relationship between the current and the displayed voltage, set the ratio to 1 mV/mA in the **Current Probe Properties** dialog box.

To reverse the polarity of the **Current Probe** output, right-click on the **Current Probe** and select **Reverse Probe Direction** from the pop-up that appears.

LabVIEW Instruments

Note This section describes the LabVIEW instruments functionality available in Multisim 10.1. For information about LabVIEW instruments in previous versions of Multisim, refer to the documentation included with that software.

Note LabVIEW instruments created for Multisim 9 and 10 will continue to function in Multisim 10.1 but will not have any of the new functionality described below. You must reinstall any existing LabVIEW instruments in Multisim 10.1 using the instructions in the *Installing a LabVIEW Instrument* section.

You can create your own custom instruments using the LabVIEW graphical development environment. Instruments that you create using LabVIEW can take advantage of the full functionality of the LabVIEW development system, including data acquisition, instrument control, mathematical analysis, and so on. For example, you can create the following kinds of instruments:

- An instrument that acquires data from the real world using a National Instruments data acquisition device or modular instrument. Multisim will then use that data as a signal source for circuit simulation.
- An instrument that displays simulation data simultaneously with multiple measurements (running average and power spectrum, for example) made from that simulation data.

LabVIEW instruments can be either input instruments, output instruments, or input/output instruments. Input instruments receive simulation data for display or processing. Output instruments generate data to use as a signal source in simulation. Input/output instruments both receive and generate simulation data.

All LabVIEW instrument types perform their functions continuously during simulation. For example, input instruments continuously receive simulation data from Multisim during simulation. Output instruments continuously generate simulation data during simulation. Input/output instruments continuously receive and generate simulation data during simulation.

Note Output instruments in Multisim 9 and 10.0 did not have the ability to generate data continuously during simulation. Output instruments in those versions generate a finite set of data that can be repeated. To see examples of these output instruments, refer to the output instrument samples in the `. . .\samples\LabVIEW Instruments\Templates\Legacy` folder. Input/output instruments are a new capability of Multisim 10.1.

Input instruments allow the user or creator of the instrument to set a sampling rate. This sampling rate is the rate at which the instrument receives data from Multisim. This sampling rate is analogous to the sampling rate you would set for a physical data acquisition device or modular instrument that acquires data from the real world. You should observe the Nyquist sampling theorem when choosing a sampling rate for your instrument. Note that the higher the value of the sampling rate, the slower simulation will run.

Note If the sampling rate is set to zero for a LabVIEW input instrument, Multisim chooses its own sampling rate based on the simulation timestep near the start of simulation.

Input instruments also allow the user or creator of the instrument to set an interpolation method. The interpolation method is the particular algorithm Multisim uses to transform unevenly-sampled SPICE simulation data into evenly-sampled LabVIEW Waveform Data Type data. The available interpolation methods are **Coerce**, **Linear**, **Spline**, and **Force Step**. **Force Step** is not strictly an interpolation method in that it forces the Multisim SPICE simulator to take additional timesteps that line up with the requested sampling rate.

Note The **Force Step** interpolation method is new to Multisim 10.1.

If the particular LabVIEW instrument allows the user of the instrument to set the sampling rate and/or interpolation method, Multisim will recognize changes to these values during simulation.

Note In Multisim 9 and 10, Multisim ignored all changes to the sampling rate and interpolation method values during simulation.

System Requirements

To create and modify LabVIEW instruments, you must have the LabVIEW 8.2.*x* or 8.5.*x* Development System and Application Builder.

To use LabVIEW instruments, you must have the LabVIEW Run-Time Engine installed on your computer. The version of this Run-Time Engine must correspond to the version of the LabVIEW Development System used to create the instrument. NI Circuit Design Suite installs the LabVIEW 8.2.1 and 8.5.1 Run-Time Engines.

Sample LabVIEW Instruments

Multisim includes the following sample LabVIEW instruments:

- **Microphone**—Records audio from the computer sound device and outputs the sound data as a signal source.
- **Speaker**—Plays input signal using the computer sound device.
- **Signal Generator**—Generates and outputs a sine, triangle, square, or sawtooth wave.
- **Streaming Signal Generator**—Generates and outputs a sine, triangle, square, or sawtooth wave, and allows changes to the signal during simulation.
- **Signal Analyzer**—Displays time-domain data, auto power spectrum, or running average of input signal.

The source code for the sample instruments is available in the directory where you installed Multisim at `. . .\samples\LabVIEW Instruments`.

Microphone

The **Microphone** instrument allows you to record sound data from devices attached to the input of your computer's sound device (for example, microphone, CD player). It then outputs that data as a signal in Multisim. You configure the settings and record the sound before beginning simulation. Multisim will then use the audio signal as a signal source during simulation.

Complete the following steps to use the **Microphone**:

1. Place it in a schematic and open its face.
2. Select the desired audio **Device** (usually the default device is correct), the **Recording Duration**, and the desired **Sample Rate**. The higher the value you choose for the **Sample Rate**, the higher the quality of the output signal, but the slower simulation using that data will run.
3. Click **Record Sound** to record the signal that is connected to the input connection of your computer's sound device.
4. Before starting simulation, you can also choose to **Repeat Recorded Sound**. If you do not enable this option and simulate the circuit, once the simulation time has exceeded the length of the recorded signal, Multisim will continue to simulate but the output signal from the **Microphone** instrument will fall to 0 volts. If you enable this option, the **Microphone** instrument will repeatedly output the recorded data until you stop simulation.

5. Begin simulation of the circuit. The **Microphone** instrument will output the recorded sound as a voltage signal.

Speaker

The **Speaker** instrument allows you to output a voltage signal as sound using your computer's sound device. You configure the settings before beginning simulation and play the sound after stopping simulation.

Complete the following steps to use the **Speaker**:

1. Place the instrument in a schematic and open its face.
2. Select the desired audio **Device** (usually the default device is correct), the **Playback Duration**, and the **Sample Rate** at which the instrument should receive data from the simulation.

   If you are using the **Speaker** instrument in conjunction with the **Microphone** instrument, set the **Sample Rate** of the two instruments to be the same. Otherwise, set the sample rate of the Speaker to be at least twice the frequency of the input signal. Note that the higher the value of the sample rate, the slower simulation will run.
3. Begin simulation of the circuit. As simulation runs, the **Speaker** instrument collects the input data until the simulation time reaches the limit you configured in the **Playback Duration** field.
4. Stop simulation and click **Play Sound** to play as sound the data that the **Speaker** stored during the simulation.

Signal Analyzer

The **Signal Analyzer** instrument is an example of how to implement a simple LabVIEW instrument that receives, analyzes, and displays simulation data. Its settings may be adjusted during simulation.

Complete the following steps to use the **Signal Analyzer**:

1. Place the instrument in a schematic and open its face.
2. Set the desired **Analysis Type** and **Sampling Rate** (rate at which the instrument receives data from the simulation). The sampling rate of the **Signal Analyzer** should be at least twice the frequency of the input signal.
3. Begin simulation.
4. Optionally, change the **Sampling Rate** or **Interpolation Method** during simulation.

Signal Generator

The **Signal Generator** instrument is an example of how to implement a simple LabVIEW instrument that generates or acquires data and then outputs it as a signal source for simulation.

This instrument only uses functionality available in Multisim 9 and 10. It is provided for compatibility with circuits created in those versions with the existing **Signal Generator** instrument.

Complete the following steps to use the **Signal Generator**:

1. Place the instrument in a schematic and open its face.
2. Set the desired **Signal Information** parameters and **Sampling Info**. If desired, enable **Repeat Data**.
3. Begin simulation. The instrument generates the output data and then outputs that data as a signal source for the simulation.

Streaming Signal Generator

The **Streaming Signal Generator** instrument is an example of how to implement a simple LabVIEW instrument that generates data and continuously outputs it as a signal source in Multisim.

Complete the following steps to use the **Streaming Signal Generator**:

1. Place the instrument in a schematic and open its face.
2. Set the desired **Signal Information** parameters and **Sampling Info**.
3. Begin simulation. Multisim continuously requests output data from the instrument during the simulation.

   If you adjust the **Signal Information** parameters during simulation, the data streaming out to Multisim is adjusted.

LabVIEW Instruments Toolbar

The **LabVIEW Instruments** toolbar buttons are described below.

Button	Description
	Microphone button. Places a LabVIEW microphone on the workspace. Refer to the *Microphone* section for more information.
	Speaker button. Places a LabVIEW speaker on the workspace. Refer to the *Speaker* section for more information.
	Signal Analyzer button. Places a LabVIEW signal analyzer on the workspace. Refer to the *Signal Analyzer* section for more information.
	Signal Generator button. Places a LabVIEW signal generator on the workspace. Refer to the *Signal Generator* section for more information.
	Streaming Signal Generator button. Places a LabVIEW streaming signal generator on the workspace. Refer to the *Streaming Signal Generator* section for more information.

Creating a LabVIEW Instrument

Note This section assumes existing familiarity with LabVIEW and concepts such as VIs, VI Libraries, Projects, and Build Specifications. Refer to the LabVIEW documentation for more information on these topics.

Note You can find templates for the LabVIEW instruments from Multisim 9 and 10 in the `...\samples\LabVIEW Instruments\Templates\Legacy` folder.

The primary component of a LabVIEW instrument for Multisim is a VI template (`.vit` file) that serves as the face of the instrument and handles communication with Multisim.

Multisim includes templates for creating input, output and input/output instruments. These starter templates include a LabVIEW Project with the necessary settings for building the final instrument and a VI template that contains the front panel and block diagram objects necessary for the VI to communicate with Multisim.

These starter templates are available in the `samples` directory at:

- `...\samples\LabVIEW Instruments\Templates\Input` – Use these files when creating a new instrument that receives simulation data from Multisim.
- `...\samples\LabVIEW Instruments\Templates\Output` – Use these files when creating a new instrument that generates data to use as a signal source in Multisim.
- `...\samples\LabVIEW Instruments\Templates\InputOutput` – Use these files when creating a new instrument that both receives simulation data from Multisim and generates data to use as a signal source in Multisim.

The starter Projects, `StarterInputInstrumentV2.lvproj`, `StarterOutputInstrumentV2.lvproj`, and `StarterInputOutputInstrumentV2.lvproj` each contain a Source Distribution Build Specification. The output of each Build Specification will go to:

`...\samples\LabVIEW Instruments\Templates\Input\Build`, `...\samples\LabVIEW Instruments\Templates\Output\Build`, and `...\samples\LabVIEW Instruments\Templates\InputOutput\Build`.

This output is what you must install into Multisim to use the instrument. Refer to the *Building a LabVIEW Instrument* and *Installing a LabVIEW Instrument* sections for information about building and installing a LabVIEW instrument.

Complete the following steps to create your own Multisim instrument:

1. Copy the contents of the `...\samples\LabVIEW Instruments\Templates\Input`, `...\samples\LabVIEW Instruments\Templates\Output`, or `...\samples\LabVIEW Instruments\Templates\InputOutput` directory to an empty directory on your computer.
2. Open the copy of `StarterInputInstrumentV2.lvproj`, `StarterOutputInstrumentV2.lvproj`, or `StarterInputOutputInstrumentV2.lvproj` in LabVIEW.
3. In the Project tree, right-click on **My Computer»Instrument Template»Starter Input Instrument V2.vit**, **Starter Output Instrument V2.vit**, or **Starter IO InstrumentV2.vit** and select **Open**.

4. In the window for the VI template, select **File»Save As**. In the save dialog that appears, select the **Rename** option and click **Continue**. In the next dialog, choose a new name (and location, if you desire) for the VI template and click **OK**.
5. Close the newly renamed VI template.
6. In the **Project** tree, right-click: **My Computer»subVIs»Starter Input Instrument V2_multisimInformation.vi**, **Starter Output Instruments V2_multisimInformation.vi**, or **Starter IO Instrument V2_multisimInformation.vi** and select **Open**.
7. In the window for the VI, select **File»Save As**. In the save dialog that appears, select the **Rename** option and click **Continue**. In the next dialog, choose a new name (and location, if you desire) for the VI and click **OK**. The new name for the VI should be: `<Name chosen in step 4, minus the .vit extension>_multisimInformation.vi`. For example, if you renamed the VI template in step 4 `My Instrument.vit`, the name for this VI should be `My Instrument_multisimInformation.vi`.
8. Close the newly renamed VI.
9. Right click on the **My Computer»Build Specification»Source Distribution** item in the Project tree and select **Properties** from the context menu to edit the Build Specification.
10. In the **Custom Destinations** section of the **Destination Settings** page of the **Build Specification Properties** dialog, modify the final VI library in the **Destination Path** for the **Destination Directory** to something unique (for example, `My Custom Instrument.llb`).
11. In the same location, modify the final directory in the **Destination Directory** for the **Support Directory** to something unique (for example, `My Custom Instrument`).
12. Click **OK** on the **Build Specification Properties** dialog and save the Project file.
13. Open the template VI.
14. Edit the icon of the template VI. Multisim will use this icon in the center of the symbol for the instrument.
15. Follow the guidelines and instruments listed in the comment fields on the block diagram of the template VI to create your custom instrument. The sample LabVIEW instruments provide excellent guides for how to modify the starter instruments to add custom functionality.
16. Save the VI template.

Building a LabVIEW Instrument

To ready a LabVIEW instrument for installation into Multisim, build the Source Distribution Build Specification in the LabVIEW Project for your instrument.

To build the Build Specification for your instrument, right-click the **My Computer»Build Specification»Source Distribution** item in the Project tree for the instrument and select **Build** from the context menu.

The output of this build process is as follows:

- `VI library file (.llb)` containing the main VI template, any VIs used in the hierarchy of the main VI template, and all VIs that are members of Libraries referenced in the main VI template hierarchy, regardless of whether they are actually used in the hierarchy of the instrument.
- A directory with the same name as the VI library (minus the `.llb` extension) that contains any non-VI portions of the template hierarchy and referenced Projects. These files can be DLLs, LabVIEW menu files, and so on.

Installing a LabVIEW Instrument

To install a LabVIEW instrument into Multisim, copy the output of the build process (a uniquely named VI library and directory) into the location specified in the **User LabVIEW Instruments Path** setting found in the **Paths** tab of Multisim's **Preferences** dialog box.

The next time you launch Multisim, the instrument will appear under the **LabVIEW Instruments** button in the **Instruments** toolbar.

Guidelines for Successfully Creating a LabVIEW Instrument

When creating a LabVIEW instrument for Multisim, you must follow certain guidelines:

- Always create new instruments from either the template instrument files or one of the example instruments. These files include the front panel objects, block diagram objects, and settings necessary for the instrument to behave correctly.
- Do not delete or modify the existing diagram objects in the starter template VI. You may add additional controls/indicators and handle additional events in the main event loop, but do not modify or delete any of the existing event-handling code.

- You may add diagram code to the starter template VI where specified in the comments on the block diagram.
- Every LabVIEW instrument installed in Multisim must have a unique name. Specifically, the VI library that contains the main VI template, the support files directory, and the main VI template itself must all have unique names.
- All subVIs used in a LabVIEW instrument must have unique names unless you intend to use the same subVI across multiple instruments.
- All Libraries used in a LabVIEW instrument must have unique names unless you intend to use the same Libraries across multiple instruments.
- All LabVIEW instruments that contain VIs that are part of a Library must be built using the same version of that Library. If you build an instrument using a Library and then make changes to that Library before building another instrument to install on the same computer, you must also rebuild and reinstall the original instrument.
- The Source Distribution Build Specification in the LabVIEW Project for the instrument must be configured to always include all contained items. To make this configuration, navigate to the Source File Settings page of the Source Distribution Build Specification Properties dialog, select Dependencies in the Project Files tree, and enable the **Set inclusion type for all contained items»Always include option**. This option is already enabled correctly in the starter instrument LabVIEW Projects.
- Consider whether subVIs you create for your instruments should be marked as Reentrant. If subVIs use instance-specific constructs such as uninitialized shift registers, the First Call? function, and so on, you will need to mark the VI for **Reentrant execution** under **File»VI Properties»Execution**. This setting is necessary to allow multiple instances of the same LabVIEW instrument to behave correctly.
- When creating an output instrument that generates data continously, you must remember that, in general, Multisim simulates circuits at a rate slower than the real-time behavior of an equivalent physical circuit. Therefore, if you were to create an instrument that continuously acquired real-world data using a data acquisition device, modular instrument, or other such hardware, and use that data as a simulation source in Multisim, simulation would not be able to keep up with the acquired data because it is running at a rate slower than the data acquisition. It is the responsibility of the output instrument creator to generate sufficiently small chunks of data at a rate that allows Multisim simulation to keep up with the data acquisition.

- Input/output instruments have all of the capabilities of input-only and output-only instruments. Note, however, that LabVIEW instrument input data is *lagging* (meaning that it arrives at the instrument in discrete chunks after the simulator has produced the data), and streaming out data is *leading* (meaning that the simulator requests the data in discrete chunks before it needs it). This behavior means that tight feedback loops where a LabVIEW instrument's output depends instantaneously on its intput are not possible. In general, it will take at least one extra set of continuous outputs through the "Update Data Get Output Values" event of the instrument before you can be sure that the input received by that instrument reflects an adjusted output signal.

10

Analyses

This chapter explains how to use the various analyses included in Multisim. It explains how to work with analyses in general, the specific settings and options for each individual analysis, and how to view and manipulate analyses results.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Introduction to Multisim Analyses

Multisim offers you many analyses, all of which use simulation to generate the data for the analysis you want to perform. These analyses can range from quite basic to extremely sophisticated, and can often require one analysis to be performed (automatically) as part of another.

For each analysis, you will need to decide upon some settings that will tell Multisim what you want the analysis to do. In addition to the analyses provided by Multisim, you can also create user-defined analyses based on SPICE commands you enter.

When you activate an analysis, the results are displayed on a plot in the Multisim **Grapher**, unless you specify otherwise, and saved for use in the **Postprocessor**. Refer to the *Viewing the Analysis Results: Grapher* section and Chapter 11, *Postprocessor* for more information.

Some results are also written to a viewable audit trail. Refer to the *Simulation Error Log/Audit Trail* section of Chapter 8, *Simulation*, for more information.

To indicate that an analysis is running, the **Simulation Running Indicator** appears in the status bar as shown in the figure below. This indicator flashes until the analysis is complete.

Note Some analyses are performed so quickly, that you will not notice the **Simulation Running Indicator** flashing.

Viewing the Analysis Results: Grapher

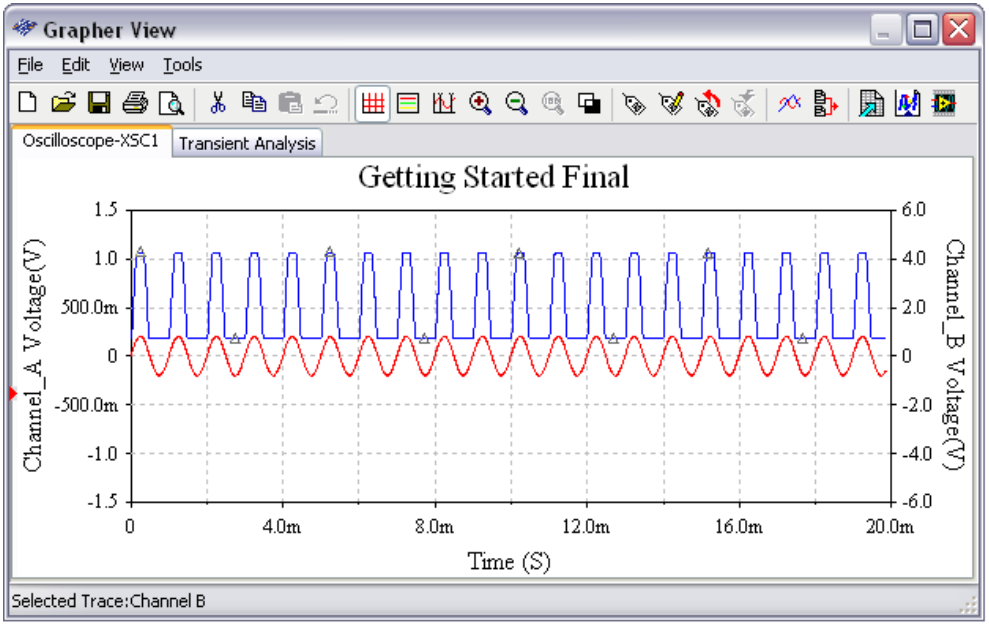
To have the **Grapher** appear, select **View»Grapher**.

The **Grapher** is a multi-purpose display tool that lets you view, adjust, save, and export graphs and charts. It is used to display:

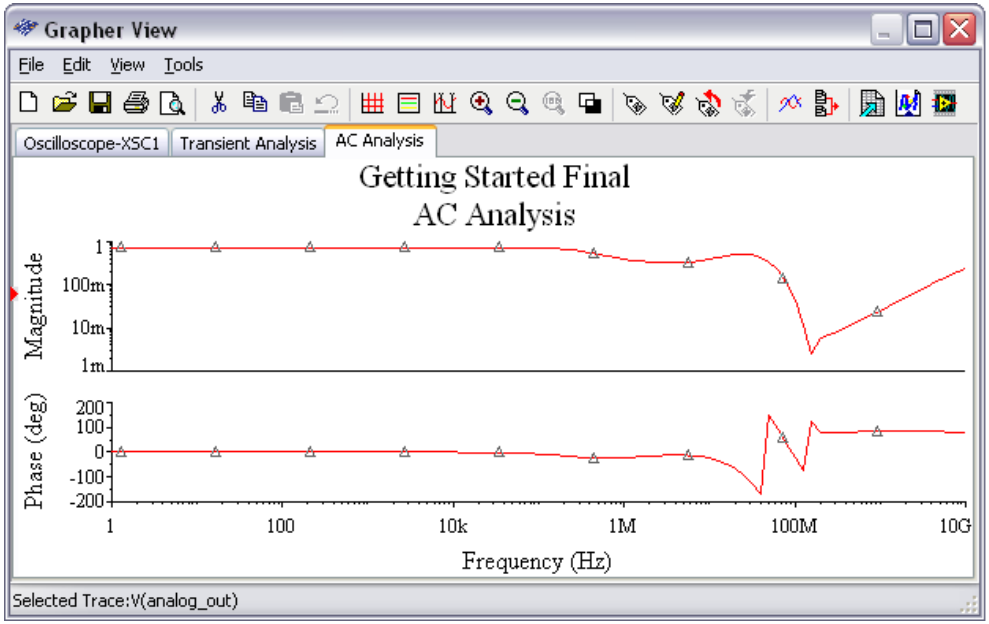
- the results of all Multisim analyses in graphs and charts
- a graph of traces for some instruments (for example the results of the postprocessor, oscilloscope and Bode plot).

The display shows both graphs and charts. In a graph, data are displayed as one or more traces along vertical and horizontal axes. In a chart, text data are displayed in rows and columns. The window is made up of several tabbed pages, depending on how many analyses, etc. have been run.

Each page has two possible active areas which will be indicated by a red arrow: the entire page indicated with the arrow in the left margin near the page name of the chart/graph indicated with the arrow in the left margin near the active chart/graph. Some functions, such as cut/copy/paste, affect only the active area, so be sure you have selected the desired area before performing a function.



Some analyses, such as the **AC Analysis**, create both a phase and magnitude graph on the same page when the analysis is run. In this case, the x-axes (frequency) of the plots will line up as shown in the figure below. This is the default setting. However, you can change the setting for individual graphs. Refer to the *Axes* section for more information.



The **Grapher** offers a number of functions on its toolbar:

Button	Description
	Opens a new page tab.
	Opens a dialog where you select a previously saved graph file to open.
	Saves contents of current pages in a graph file.
	Prints some or all of the current pages. Refer to the *Print and Print Preview* section for more information.
	Shows preview of pages as they will be printed. Refer to the *Print and Print Preview* section for more information.
	Cuts pages or graphs/charts. Refer to the *Cut, Copy and Paste* section for more information.

Button	Description
	Copies pages or graphs/charts. Refer to the *Cut, Copy and Paste* section for more information.
	Pastes pages or graphs/charts. Refer to the *Cut, Copy and Paste* section for more information.
	Undoes most recent action.
	Applies/removes grid for the selected graph. Refer to the *Grids and Legends* section for more information.
	Displays/hides graph legend. Refer to the *Grids and Legends* section for more information.
	Displays/hides cursors and data. Refer to the *Cursors* section for more information.
	Zoom settings. Refer to the *Zoom and Restore* section for more information.
	Displays/hides negative image of graph or chart.
	Refer to the *Working with Pages on the Grapher* section for information.
	Refer to the *Working with Graphs* section for information.
	Copies graph properties.
	Pastes graph properties.

Button	Description
	Adds traces from the latest simulation results.
	Overlay traces.
	Lets you export analysis results to Excel. Refer to the *Exporting to Excel* section of Chapter 13, *Transfer/Communication*, for more information.
	Lets you export analysis results to MathCAD. Refer to the *Exporting to MathCAD* section of Chapter 13, *Transfer/Communication*, for more information.
	Lets you export analysis results to either a `.lvm` or `.tdm` measurement file. Refer to the *Saving to a Measurement File* section of Chapter 13, *Transfer/Communication*, for more information.

Working with Pages on the Grapher

Every analysis you perform on a circuit displays its results on a separate page of the **Grapher**. Every trace may also appear on a separate page, if that is how you have set up your analysis.

To view a page on the **Grapher**, click its tab.

To scroll through pages (when there are too many tabs to fit in the available space), click the forward or reverse arrow buttons that appear at the right edge of the tabs.

Complete the following steps to change page properties:

1. Select a page by clicking its tab.
2. Click the **Page Properties** button to display the **Page Properties** dialog box.
3. Change the following as desired:
   - **Tab Name** field—Changes the tab's name.
   - **Title** field—Changes the title of the chart or graph.
   - **Font** button—Changes the title's font.
   - **Background Color** drop-down list—Changes the page's background color.

- **Show/Hide Diagrams on Page** button—Displays a list where you select which diagrams to display on the page.

4. To apply the change and close the dialog box, click **OK**. To apply the change and leave the dialog box open for additional selections, click **Apply**.

Working with Graphs

To help you examine graphical data, you can use a grid, a legend and vertical cursors. You can also zoom in on any part of a graph. You can apply these tools separately or together. In addition, you can change several graph display characteristics from the tabs of the **Graph Properties** dialog box.

Note To display the **Graph Properties** dialog box or to use the buttons described in this section, you must have a graph selected. Click on a graph to select it. A red arrow appears to the left of the graph to indicate it is selected. Your selection is also indicated in the **Status Bar** at the bottom of the **Grapher**.

Grids and Legends

Complete the following steps to apply a grid to a graph:

1. Select a graph by clicking anywhere on it.
2. Click the **Show/Hide Grid** button. To remove the grid, click the button again.

Or

1. Select a graph by clicking anywhere on it.
2. Click the **Properties** button. The **Graph Properties** dialog box appears.

3. Click the **General** tab.
4. Enable the **Grid On** option. If desired, change the grid **Pen Size** and color.

Complete the following steps to apply a legend to a graph:

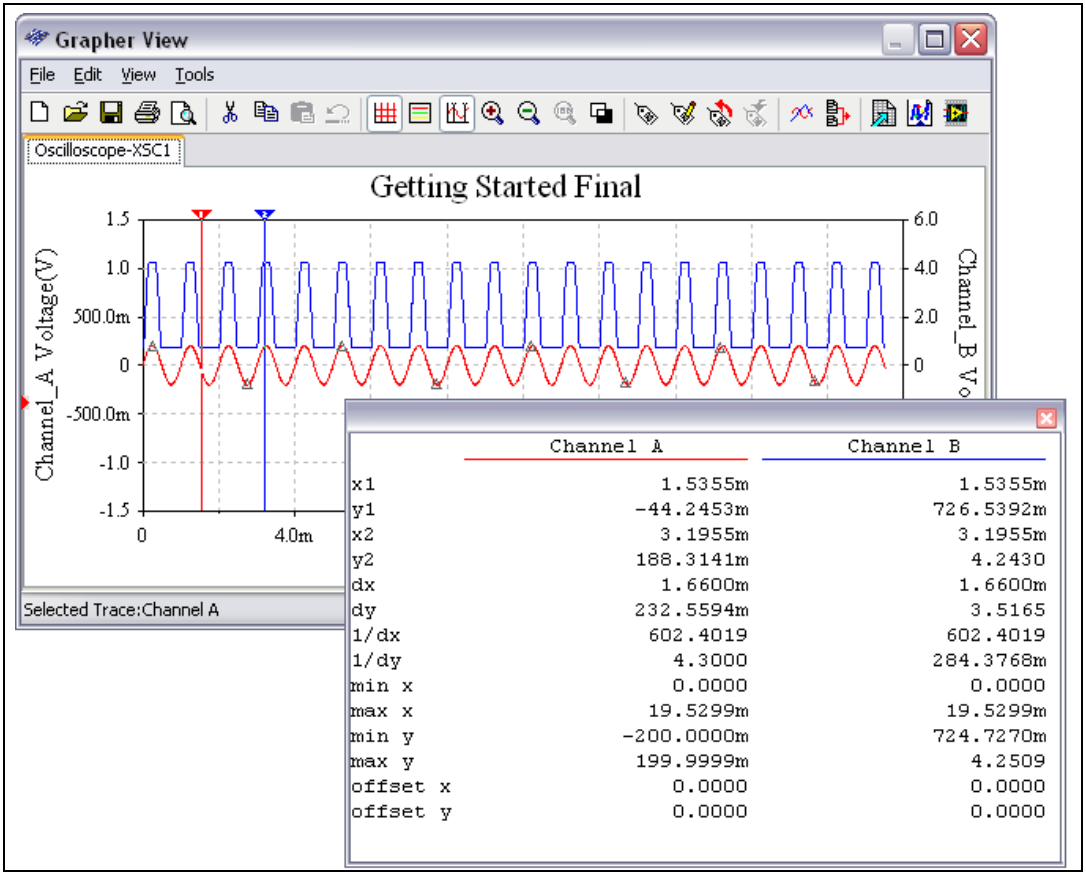
1. Select a graph by clicking anywhere on it.
2. Click the **Show/Hide Legend** button. To remove the legend, click the button again.

Or

1. Select a graph by clicking anywhere on it.
2. Click the **Properties** button. The **Graph Properties** dialog box appears.
3. Select the **General** tab.
4. Enable the **Legend On** option. If desired, change the labels of the traces using the **Traces** tab. Refer to the Traces section for more information.

Cursors

As shown in the figure below, when you activate the cursors feature, two vertical cursors appear on the selected graph. At the same time, a window pops up, displaying a list of data for one or all traces.



	Channel A	Channel B
x1	1.5355m	1.5355m
y1	-44.2453m	726.5392m
x2	3.1955m	3.1955m
y2	188.3141m	4.2430
dx	1.6600m	1.6600m
dy	232.5594m	3.5165
1/dx	602.4019	602.4019
1/dy	4.3000	284.3768m
min x	0.0000	0.0000
max x	19.5299m	19.5299m
min y	-200.0000m	724.7270m
max y	199.9999m	4.2509
offset x	0.0000	0.0000
offset y	0.0000	0.0000

The cursor data include:

- **x1,y1**—(x,y) co-ordinates for the left cursor
- **x2,y2**—(x,y) co-ordinates for the right cursor
- **dx**—X-axis delta between the two cursors
- **dy**—X-axis delta between the two cursors
- **1/dx**—Reciprocal of the x-axis delta
- **1/dy**—Reciprocal of the y-axis delta
- **min x, min y**—x and y minima within the graph ranges
- **max x, max y**—x and y maxima within the graph ranges

Complete the following steps to activate the cursors:

1. Select a graph by clicking anywhere on it.
2. Click the **Show/Hide Cursors** button. To hide the cursors, click the button again.

Or

1. Select a graph by clicking anywhere on it.
2. Click the **Properties** button. The **Graph Properties** dialog box appears.

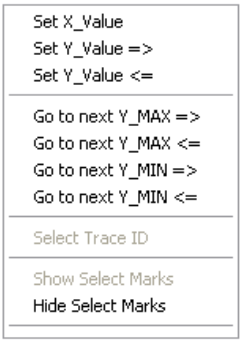
3. Select the **General** tab.
4. Enable the **Cursors On** option.
5. Select **Single Trace** to view cursor data for one trace or **All Traces** to view cursor data for all traces. If you select **Single Trace** and there is more than one trace in your graph, use the **Trace** field to select the one you want.

To move a cursor, click and drag it horizontally.

Cursor Pop-up Menu

Complete the following steps to move a cursor to a precise location:

1. Right-click on the cursor you wish to move. The cursor pop-up displays as shown below.



2. Choose **Select Trace ID**. The following dialog box appears.

3. Select the trace which you wish to use as the reference for the other options in the pop-up menu from the **Trace** drop-down list. When you select any of the other options in the pop-up, the cursor will move to the specific location on the trace that you select in this step.
4. Right-click on the cursor you wish to move and choose one of the following options:
   - **Set X_Value**—Click to display the following dialog box. Enter the desired location on the x-axis and click **OK**. The cursor moves to that location.
   - **Set Y_Value =>**—Click to display the following dialog box. Enter the desired location on the y-axis to where you would like the cursor to move and click **OK**. The cursor moves to the *right*, to the first place when that value occurs.
   - **Set Y_Value <=**—Click to display the following dialog box. Enter the desired location on the y-axis to where you would like

the cursor to move and click **OK**. The cursor moves to the *left*, to the first place when that value occurs.

- **Go to next Y_MAX =>**—Click to move the cursor to the first occurence of the maximum Y value that is to the *right* of its current location.
- **Go to next Y_MAX <=**—Click to move the cursor to the first occurence of the maximum Y value that is to the *left* of its current location.
- **Go to next Y_MIN =>**—Click to move the cursor to the first occurence of the minimum Y value that is to the *right* of its current location.
- **Go to next Y_MIN <=**—Click to move the cursor to the first occurence of the minimum Y value that is to the *left* of its current location.

Zoom and Restore

Complete the following steps to zoom in on any part of a graph:

1. Select a graph by clicking anywhere on it.
2. Click and drag the pointer until the dotted enlargement box covers the region of the graph that you want to zoom in on.
3. Release the mouse button. The axes are scaled and the graph redrawn based on the enlargement box.

Or

1. Select a graph by clicking anywhere on it.

2. Click the **Properties** button. The **Graph Properties** dialog box appears.
3. Click an axis tab to zoom along that axis. For example, choose the **Bottom Axis** tab to zoom along the horizontal dimension. (Check the **Traces** tab to see which axis is used for the range you want to zoom.)
4. Type a new minimum and maximum in the **Range** box.

To restore a graph to its original scale, click the **Zoom Restore** button.

Title

Complete the following steps to apply a title to a graph:

1. Select a graph by clicking anywhere on it.

2. Click the **Properties** button. The **Graph Properties** dialog box appears.

3. Choose the **General** tab.
4. Type a new title in the **Title** field. To change the title's font, click the **Font** button.

Axes

You can change several characteristics of a graph's axes from the four axes tabs in the **Graph Properties** dialog box. The options are identical in each of the tabs.

Complete the following steps to change the characteristics of an axis:

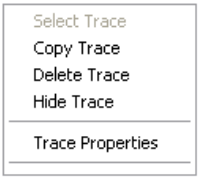
1. Select a graph by clicking anywhere on it.
2. Click the **Properties** button. The **Graph Properties** dialog box appears.
3. Click the **Axis** tab for the axis you want to change.
4. Change any of the axis' characteristics, using the following fields:
   - **Label**—Label for the axis. To change the axis font, click **Font**.
   - **Pen Size**—Controls the thickness and color of the axis and the font of the axis' values. To change color or font, click **Color** or **Font**.
   - **Minimum/Maximum**—Controls the minimum and maximum values displayed. Values change when you zoom.
   - **Number**—Sets the number of tick marks on axis.
   - **Frequency**—Sets the occurrence of values on tick marks. For example, "2" means that a value appears every two tick marks.
   - **Precision**—Sets the number of significant digits for axis values.
   - **Scale**—Sets a multiplication factor for axis values. Changes the scale of the axis.
   - **Enabled**—Determines whether or not the axis appears.

Traces

To select a trace, left-click on it. Trace select marks appear on the selected trace (small triangles along the trace.

Complete the following steps to change other trace properties:

1. Right-click on the trace to display the following pop-up.



2. Select the desired item from the pop-up.

You can change several characteristics of each trace in a graph from the **Traces** tab in the **Graph Properties** dialog box.

Complete the following steps to change the characteristics of a trace:

1. Select a graph by clicking anywhere on it.
2. Click the **Properties** button. The **Graph Properties** dialog box appears.
3. Click the **Traces** tab.
4. Select a trace.
5. Change any of the trace's characteristics, using the following fields:
   - **Trace**—Specifies the trace whose properties are being affected.
   - **Label**—Specifies a label for the trace. Appears in legend.
   - **Pen Size**—Controls the thickness of the trace.
   - **Color**—Controls the color of the trace. The Sample box shows a preview.
   - **Bottom Axis/Top Axis**—Controls the X range of the trace.
   - **Left Axis/Right Axis**—Controls the Y range of the trace.
   - **X Offset/Y Offset**—Value to offset the trace from its original coordinates.
   - **Auto-Separate** button—Offsets multiple traces in a graph for easier viewing.

Merging Traces

Complete the following steps to traces together onto a new graph:

1. Click on the **Overlay Traces** button. The **Select a Graph** dialog box appears.
2. Select the files you wish to merge together and click **OK**. A new graph appears with the merged traces.

Select Pages dialog box

The **Select Pages** dialog box is used to select pages to print, preview or remove from the **Grapher**.

Note The **Select Pages** dialog does not appear if there is only one page open in the **Grapher**.

Complete the following steps to preview pages before printing:

1. Select **File»Print Preview** from the **Grapher**. The **Select Pages** dialog box appears.
2. Check the pages you wish to preview and click **OK**.

Complete the following steps to print pages:

1. Select **File»Print** from the **Grapher**. The **Select Pages** dialog box appears.
2. Check the pages you wish to print and click **OK**.

Complete the following steps to remove pages from the **Grapher**:

1. Select **Edit»Clear Pages**. The **Select Pages** dialog box appears.
2. Check the pages you wish to delete and click **OK**.

Graph Pop-up Menu

Complete the following steps to view the graph's pop-up menu:

1. Right-click on the background of a graph to display the graph pop-up.

Note Refer to the *Traces* section for the pop-up that is accessed from a trace.

2. Select the desired option from the pop-up.

Viewing Charts

To help you examine and organize a chart, you can sort rows, adjust column widths, change precision and add a title.

To sort a row of data, click the column name button of the column you want to sort by. Sorting order is from low to high for numbers; otherwise, it is alphabetical.

To adjust the width of a column, click and drag the left edge of the column name button.

Complete the following steps to change the chart's column precision (number of significant digits) or title:

1. Select a chart by clicking anywhere on it.
2. Click the **Properties** button. The **Chart Properties** dialog box appears.
3. To change the chart **Title**, type a new title. To change the font, click the **Font** button.

4. To change a column's precision, select a column number and a precision (number of significant digits). Precision only affects columns that contain numerical values.
5. Click **OK**.

Cut, Copy and Paste

The **Grapher** lets you cut, copy and paste pages, graphs and charts.

Note You must use the cut, copy and paste buttons from the **Grapher** window. You cannot use the Multisim menus, buttons or keyboard shortcuts for these functions.

Complete the following steps to cut, copy and paste pages:

1. Select a page by clicking on its tab.
2. Click the **Cut** or **Copy** button.

3. Click the **Paste** button. The cut or copied page appears.

Note When a page is selected (the red arrow points to the tab), cut, copy and paste affect page properties only. They do *not* affect the graphs or charts on the page.

Complete the following steps to cut, copy and paste graphs and charts:

1. Select a graph or chart.
2. Click the **Cut** or **Copy** button.
3. Click the **Paste** button to paste the graph or chart onto the same page.

   Or, to paste onto a new page:

4. Click the **New** button. A new tab appears in the **Grapher**.
5. Click the **Paste** button.

Note When a graph or chart is selected (the red arrow points to the graph or chart), cut, copy and paste affect the selected graph or chart only. They do *not* affect overall page properties.

Opening Files

Complete the following steps to open an existing file in the **Grapher**:

1. Click the **Open** button. A file browser appears.
2. Select the desired file type from the drop-down list. Available file types are:
   - **Graph files** (`*.gra`)—Multisim **Grapher** files.
   - **Data files** (`*.dat`)—Files created and saved from the **Agilent Simulated Oscilloscope**.
   - **Scope data** (`*.scp`)—Files created and saved from the Multisim **Oscilloscope**.
   - **Bode data** (`*.bod`)—Files created and saved from the Multisim Bode plotter.
3. Select the file you want to open.
4. Click **Open**.

Saving Files

Simulation data in Multisim consists of unevenly sampled time-value pairs. This is not an issue when saving either Multisim **Grapher** files (`*.gra`) or standard text files (`*.txt`). However, text-based measurement files (`*.lvm`) such as those created in NI LabVIEW require that the data be evenly sampled. Depending on how you wish to use the data, you may also wish to resample the data for binary measurement files (`*.tdm`) so that they are evenly sampled.

Details follow.

Complete the following steps to save a file in the **Grapher**:

1. Select **File»Save As**. A file browser appears.
2. Select the desired file type from the drop-down list. Available file types are:
   - **Graph files** (`*.gra`)—Multisim **Grapher** files.
   - **Text files** (`*.txt`)—Standard text files.
   - **Text-based measurement files** (`*.lvm`)—Files such as those created in NI LabVIEW.
   - **Binary measurement files** (`*.tdm`)—Files used to exchange data between National Instruments software, such as NI LabVIEW and DIAdem. Note that when you save data this file type, two files are created; a header file (`.tdm`) and a binary file (`.tdx`). Refer to the

Important Information about Saving Simulation Data to LVM and TDM Files section for more information.

3. Select the desired filepath, enter a filename and click **Save**. Graph files (`*.gra`) and text files (`*.txt`) are saved to the location specified.

   If you selected either text-based measurement files (`*.lvm`) or binary measurement files (`*.tdm`), the **Data resampling settings** dialog box appears.

4. If you chose to save the file as a `.lvm` or `.tdm` file, the **Data resampling settings** dialog box appears. Change the following settings as desired:

   - **Do not resample** checkbox—Appears for `.tdm` files only. Enable if you do not wish to resample the data. The other options are disabled.
   - **Resample data** checkbox—Appears for `.tdm` files only.
   - **Interpolation mode**—select one of: **Coerce**; **Linear Interpolation**; **Spline Interpolation**. (See below for a description of each).
   - **Δx (in seconds if time-domain data)**—The sampling period to use for resampling.
   - **1/Δx (in Hz if time-domain data)**—The sampling rate to use for resampling.

Note You can change either **Δx** or **1/Δx**. The other will change accordingly.

   - **Estimated file size**—This read-only field changes as you change either **Δx** or **1/Δx**.

5. Click **OK** to close the dialog and save the file.

Interpolation Modes

Coerce—This method returns an output sample value equal to the input sample value closest to the output sample value in time. An advantage of this method is that it requires almost no computation.

Linear Interpolation—This method assumes that you know two neighboring samples of the waveform, and that between those two samples the signal changes at a constant rate. The algorithm essentially draws a straight line between these two samples and returns the appropriate point along that line.

Spline Interpolation—This method uses a cubic spline algorithm, which yields smooth transitions between the samples and good quality on single-shot short records.

Important Information about Saving Simulation Data to LVM and TDM Files

In some instances, you may wish to save results with multiple graphs on the same page to a TDM or LVM file. For example, if you run an AC analysis, both a magnitude and a phase graph appear on the same **Grapher** page.

Multisim saves TDM files with each Multisim graph worth of data as a Channel Group. The Channel Group contains all data channels (voltage, current, etc.) as well as a channel describing the x-axis (usually time or frequency). You cannot use the LabVIEW Read From Measurement File Express VI to read these files. Rather, you must use the **File I/O/Storage** VIs to read the files.

If you write multiple graphs of data to an LVM file, the data for the different graphs are stored in different sections of the LVM file. To read all sections from an LVM file, you must call the LabVIEW Read from Measurement File Express VI multiple times until the end of file output is False.

Print and Print Preview

Complete the following steps to view the printed pages before you print:

1. Click the **Print Preview** button. The **Select Pages** dialog box appears.
2. Select the page that you wish to view and click **OK**.
3. Click the **Print** button to open the print dialog box and print the pages.

   Or

   Click **Close** to close print preview.

Complete the following steps to print pages:

1. Click the **Print** button on the toolbar or from the print preview. The **Select Pages** dialog box appears.
2. Select the page that you wish to print and click **OK**. The **Print** dialog box for your printer appears.
3. Enter the desired parameters and click **OK**.

Note Colored lines are distinguished with different line styles for black and white printers.

Adding Traces from the Latest Simulation Results

When you run an analysis, the results (based on the items in the **Selected variables for analysis** column of the analysis's **Output** tab) appear in the **Grapher**.

Multisim also lets you plot extra traces from the results of the most recent simulation in the **Grapher**. Before running your analysis, you simply select the variables that you want to save with the simulation. These variables are then available for use in the **Grapher** and the **Postprocessor**.

Note This section provides instructions on how to add traces from the most recent simulation using the **Grapher**. In most cases, this will likely be your method of choice. However, if you would like to use the **Postprocessor** to add traces and charts from multiple sets of simulation results, refer to the *Using the Postprocessor* section of Chapter 11, *Postprocessor*, for more information.

Complete the following steps to select variables to save for later use:

1. Click **Select variables to save** from the **Output** tab of the desired analysis to display the **Select Variables to Save** dialog box.
2. In each of the **Voltages** and **Currents** drop-down lists, select one of: **All**; **All except submodules**; **Only at static probes**; **None**.
3. In the **Power** drop-down list, select one of: **All**; **All except submodules**; **None**.
4. Click **OK** to return to the analysis's **Output** tab.
5. Run the analysis. The results (based on the items in the **Selected variables for analysis** column of the analysis's **Output** tab) appear in the **Grapher**. The variables that you selected to save for later use can be accessed as described below.

Complete the following steps to add traces from the latest simulation results:

1. From the **Grapher**, click the **Add traces** button to display the **Add traces from latest simulation** result dialog box.
2. Select one of:
   - **To selected graph**—Select if you want to add the new trace(s) to an existing graph. Select the correct **Page name** and **Graph name** from their respective drop-down lists.
   - **To new graph**—Select if you want to start a new graph for the new trace(s). Enter the desired **Page name** and **Graph name**.

3. From the **Variables** list, select the variable you want to include in the expression being used to define the trace, and click **Copy Variable to Expression**. The variable appears in the **Expressions selected** section.

Note To filter the **Variables** list to show only certain variables, choose from the drop-down list, for example, **All Voltages**.

4. From the **Functions** list, select the mathematical function you want included in the expression for this trace, and click **Copy function to expression**. To filter the **Functions** list to show only certain mathematical functions, choose from the drop-down list of options. Refer to the *Available Postprocessor Functions* section of Chapter 11, *Postprocessor*, for more information.

Note It is possible to manually type or modify a trace's equation although you should exercise caution if doing so.

5. Continue to choose analyses, variables and functions until the expression is complete.
6. If you wish to add another expression, click **Add** and add the desired **Variables** and **Functions** as described above. (To delete an expression from the list, select the expression and click **Delete**).
7. Click **Calculate** to display the results of the expression in the **Grapher**.

Working with Analyses

To use an analysis you should follow the general instructions for accessing and running an analysis described below. Also, each analysis has specific options that you can set:

- the analysis parameters (all have default values)
- how output variables are to be handled (required)
- a title for the analysis (optional)
- custom values for analysis options (optional).

Analysis settings are saved with the circuit.

The rest of this section describes the general procedures for performing analyses. The following sections describe the details of each analysis.

General Instructions

Complete the following steps to perform an analysis:

1. Select **Simulate»Analyses**. A menu appears with the list of analyses available.
2. Select the desired analysis. Depending on the analysis selected, the dialog box that appears will include some or all of the following tabs:
   - the **Analysis Parameters** tab, where you set the parameters for this analysis
   - the **Output** tab, where you specify what is to be done with specific analysis output (not present in all analyses).
   - the **Analysis Options** tab, where you set any custom values for analysis options
   - the **Summary** tab, where you see a consolidated view of all the settings for the analysis.

   The options and settings available in these tabs are described in the following sections.
3. To save the settings as the defaults for future use, click **Accept** in the analysis dialog box.
4. To run the simulation with the current settings, click **Simulate** in the analysis dialog box.
5. To stop the analysis, select **Simulate»Analyses»Stop Analysis**, or press `<Esc>`.
6. To run several analyses in a batch, refer to the *Batched Analyses* section.

Analysis Parameters Tab

The options available in the **Analysis Parameters** tab are different for each analysis, and are therefore described in different sections of this chapter, one per analysis. Analysis descriptions include guidelines for both normal and advanced use of the analysis.

Some lists of items are accompanied by a **Change filter** function. This lets you filter the items shown in that list, choosing whether or not to include internal nodes, submodules, open pins and device parameters.

Output Tab

This is where you choose the variables you want to use in the analysis.

- **Variables in circuit** drop-down list—Use to select types of output variables to display in the list below.
- **Filter Unselected Variables** button—Use to filter types of variables displayed.
- **Add device/model parameter** button—Use to add a parameter from a specific device or model to the list of variables.
- **Selected variables for analysis** drop-down list—Use to choose types of variables to be used in analysis.
- **Filter selected variables** button—Use to filter the variables displayed to include internal nodes, open pins and output variables.
- **Edit Expression** and **Add Expression** buttons—Use to add and edit analysis expressions.
- **Select variables to save** button—Use to select variables to save that can be used to add trace s to a graph after simulation, or for postprocessing. See the *Selecting Variables to Save* section.

For details on the functions used in this tab, see:

- *Choosing How to Handle Output Variables*
- *Filtering the Variable Lists*
- *Adding Parameters to the Variable List*
- *Selecting Variables to Save*
- *Adding Analysis Expressions*

Choosing How to Handle Output Variables

To include the output variable in the plot, select a variable from the list on the left and click **Add**.

To select more than one variable at a time, press `<Shift>` while selecting variables from the list on the left and click **Add**.

To remove an item from the right hand list, select it and click **Remove**.

Using the **Output** tab, you can also filter the variables list, filter the variables displayed, as well as add a wide range of device or model parameters.

By default, **All variables** are initially included in the **Variables in circuit** list.

Filtering the Variable Lists

Complete the following steps to filter the variables list according to general variable type:

1. Click the **Variables in circuit** drop-down list.
2. Click the general variable type (such as voltages, currents, device/model parameters) to include in the list.

Note If you have any static probes placed on the circuit, the contents of the list will default to static probes and all static probes in the circuit will appear in the list. Any probes that are designated as reference probes on the workspace will not be used as reference probes for the analysis, as reference probes are for interactive simulation only. Refer to the *Static Probe Settings* section of Chapter 9, *Instruments*, for information about static probe properties.

You can filter the variables displayed to include internal nodes (such as nodes inside a BJT model or inside a SPICE subcircuit), open pins, as well as output variables from any submodules contained in the circuit.

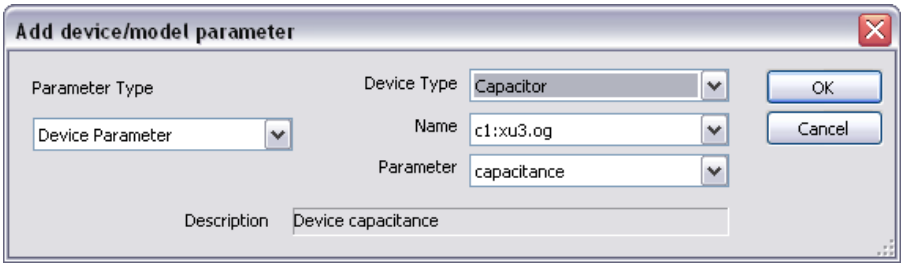
Complete the following steps to filter the variables displayed:

1. Click **Filter Unselected Variables**. The **Filter nodes** dialog box appears.
2. Enable one or more display settings.
3. Click **OK**.

Adding Parameters to the Variable List

Complete the following steps to add a parameter from a specific device or model to the list of variables:

1. Click **Add device/model parameter**. The **Add device/model parameter** dialog box appears, allowing you to specify which parameter is to be added. For example:



2. From the **Parameter Type** list, select whether you want to add a device parameter or model parameter. These let you set how various internal parameters of a component or model change during the analysis.
3. From the **Device Type** drop-down list, select a device type from the devices in the circuit.
4. From the **Name** drop-down list, select a specific instance of the device type.
5. From the **Parameter** drop-down list, select a parameter from all available device/model parameters. A brief description of the selected parameter appears in the **Description** field.
6. Click **OK** to add the selected parameter to the **Variables in circuit** list. This variable can now be included in the analysis.

The parameter appears in the left-hand list in the **Output Variables** tab on the analysis dialog box, which is automatically set to show only device/model parameters.

To delete a parameter added in this way, select it and click **Delete selected variable**.

To show the values of all the components and models in the circuit at the end of the simulation, enable **Show all device parameters at end of simulation in the audit trail**.

Selecting Variables to Save

You can select circuit variables to save that can be used to add traces to a graph after simulation, or for post-processing in the **Postprocessor**.

Complete the following steps to select variables to save for later use:

1. Click **Select variables to save** from the **Output** tab of the desired analysis to display the **Select variables to save** dialog box.
2. In each of the **Voltages** and **Currents** drop-down lists, select one of: **All**; **All except submodules**; **Only at static probes**; **None**.
3. In the **Power** drop-down list, select one of: **All**; **All except submodules**; **None**.
4. Click **OK** to return to the analysis's **Output** tab.
5. Run the analysis. The results (based on the items in the **Selected variables for analysis** column of the analysis's **Output** tab) appear in the **Grapher**. The variables that you selected to save for later use can be accessed from either the **Grapher** or the **Postprocessor**. Refer to

the *Adding Traces from the Latest Simulation Results* section and the *Using the Postprocessor* section of Chapter 11, *Postprocessor*, for information about using the variables you selected in the above procedure.

Adding Analysis Expressions

Expressions can be added to analyses via the **Analysis Expression** dialog box.

In the following analyses, this dialog box is accesed from the **Output** tab by clicking the **Add Expression** button:

- DC Operating Point Analysis
- AC Analysis
- Transient Analysis
- Fourier Analysis
- Distortion Analysis
- DC Sweep Analysis
- Parameter Sweep Analysis
- Temperature Sweep Analysis.

Other analyses support expressions through the **Analysis Parameters** tab. For the analyses listed below, you can click on the **Expression** checkbox and enter an analysis expression directly into the **Output expression** field. When the **Expression** radio button is enabled, the **Edit** button becomes active. Click this to view the **Analysis Expression** dialog box where you can enter and edit analyses.

- Sensitivity Analysis
- Worst Case Analysis
- Monte Carlo Analysis.

Complete the following steps to add an expression:

1. Access the **Analysis Expression** dialog box as described above.
2. Optionally, click **Change Filter** and select any of internal nodes, submodules or open pins to add to the **Variables** list.
3. Filter the available **Variables** and **Functions** using their respective drop-down lists.
4. Build your expression by selecting the desired **Variables** and **Functions** one at a time and click the **Copy Variable to Expression** or **Copy Function to Expression** button to place them into the

Expression field. The selection will be placed at the current cursor position.

Tip Instead of using the **Copy Variable to Expression** and **Copy Function to Expression** buttons, you can double-click to place them into the **Expression** field.

Tip To save time, you can type smaller expressions directly into the **Expression** field. (Exercise caution, as the syntax of the expression will not be checked until the analysis is performed.) If you wish to remove the entire expression and start over, click **Clear**.

Note Available functions are the same as in the **Postprocessor**, with the exception of vector functions. Vector functions require a previously acquired set of data, and are therefore available only via the **Postprocessor**. Refer to the *Available Postprocessor Functions* section of Chapter 11, *Postprocessor*, for information.

5. Click **OK** to close the dialog. Depending on the analysis from which you accessed the **Analysis Expression** dialog, your expression is placed in either the **Selected variables for analysis** list or the **Output variable** field.
6. From the analysis dialog box, click **Simulate**. The analysis proceeds as normal, and the traces on the resulting graph show the output equation(s) as if they are individual nodes.

   When you open the **Analysis Expression** dialog box again, the expression that you entered will be in the **Recent Expressions** list. Each time you add an expression in the current Multisim session, it will be added to this list.

Tip To save time entering expressions that are similar to those already used, highlight a similar expression in the **Recent Expressions** list and click **Copy to Expression**. Now edit the contents of the **Expression** field as desired. If you wish to remove an expression from the **Recent Expressions** list, highlight it and click **Delete Selected**.

Note If you run the **Postprocessor** on this analysis, the expression and all of its dependant circuit variables are available for postprocessing.

Complete the following steps to edit an expression from an analysis:

1. Highlight it in the **Selected variables for analysis** list and click **Edit Expression** to display the **Analysis Expression** dialog box.

Note If the analysis displays the expression in the **Output expression** field, there is no need to highlight it before clicking **Edit**.

2. Edit the expression as desired and click **OK** to return to the analysis.

Complete the following steps to delete an expression from an analysis:

1. Highlight the expression in the **Selected variables for analysis** list and click **Remove**. Removing an expression deletes it entirely.

   Or

   If the analysis displays the expression in the **Output expression** field, click **Edit** to display the **Analysis Expression** dialog box and click **Clear**, then **OK** to close the dialog. The **Output expression** field in the analysis is now empty.

   You can also highlight the entire expression in the **Output Variable** field, and click Delete on your keyboard.

Analysis Options Tab

The options in this tab provide you with additional flexibility, but do not require that you set them.

To check if the circuit is a valid circuit for analysis, enable **Perform consistency check before starting analysis**. This option automatically identifies inconsistencies such as open capacitors, empty circuit files and ungrounded circuits.

Normally analyses run without further intervention. If an analysis does not perform as necessary, you may need to set custom analysis options.

To change the analysis title from its default, enter text in the **Title for analysis** field.

Complete the following steps to set custom analysis options:

1. Enable **Use Custom Settings** and click **Customize**. The **Custom Analysis Options** dialog box appears.

Note You should have a general knowledge of the SPICE simulation engine before altering the default settings under this option.

2. Change the desired settings in the **Custom Analysis Options** dialog box.
3. Click **OK** to return to the **Analysis Options** tab.
4. Click **OK**.

Refer to the *Custom Analysis Options Dialog Box* section for a complete list of the available analysis options.

Summary Tab

This tab offers a quick overview of the various settings for your analysis. It does not require you to set any options, but you can use it to view summary information about your analysis.

You navigate through this display as you do with Windows Explorer. The "**+**" beside an item indicates that it has additional information underneath it, which can be revealed by clicking the "**+**". The "**–**" beside an item indicates that all its information is being revealed. That information can be hidden by clicking the "**–**".

This window also shows you the SPICE representation of your analysis options, as well as the name of the file to which the analysis results are being saved (`.raw` file). This file is used for postprocessing.

Incomplete Analyses

For a variety of reasons, the simulator in Multisim is occasionally unable to complete a simulation or an analysis.

Multisim uses the modified Newton-Raphson method to solve nonlinear circuits. When a circuit includes nonlinear components, multiple iterations of a set of linear equations are used to account for the non-linearities. The simulator makes an initial guess at the node voltages, then calculates the branch currents based on the conductances in the circuit. The branch currents are then used to recalculate the node voltages and the cycle is repeated. This cycle continues until all of the node voltages and branch currents fall within user-defined tolerances, that is, convergence occurs. You can specify tolerances and iteration limits for the analysis through the analysis options described in the *Custom Analysis Options Dialog Box* section.

If the voltages or currents do not converge within a specified number of iterations, an error message is produced and the simulation is aborted (typical messages include "Singular matrix," "Gmin stepping failed," "Source stepping failed" and "Iteration limit reached").

DC Operating Point Analysis

DC Operating Point Analysis determines the DC operating point of a circuit. The results of DC analysis are usually intermediate values for further analysis. For example, the results obtained from DC analysis determines the approximate linearized, small-signal models for any nonlinear components, such as diodes and transistors, to be used with AC frequency analysis.

Assumptions:

- AC sources are zeroed out
- Capacitors are open
- Inductors are shorted
- Digital components are treated as a large resistor to ground.

Setting up and Running DC Operating Point Analysis

To start **DC Operating Point Analysis**, select **Simulate»Analyses» DC Operating Point**.

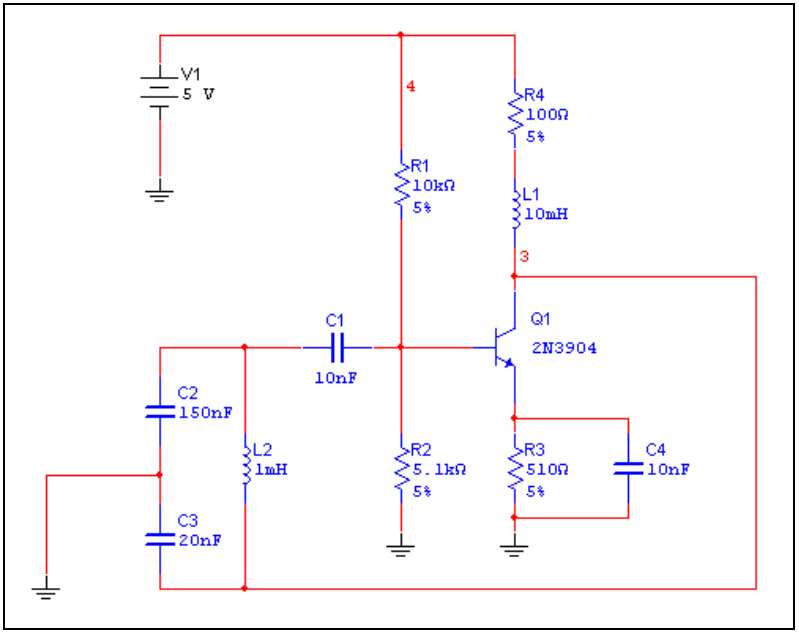
Setting DC Operating Point Analysis Parameters

There are no analysis parameters to be set for this analysis. Hence, the **Analysis Parameters** tab does not appear for **DC Operating Point Analysis**.

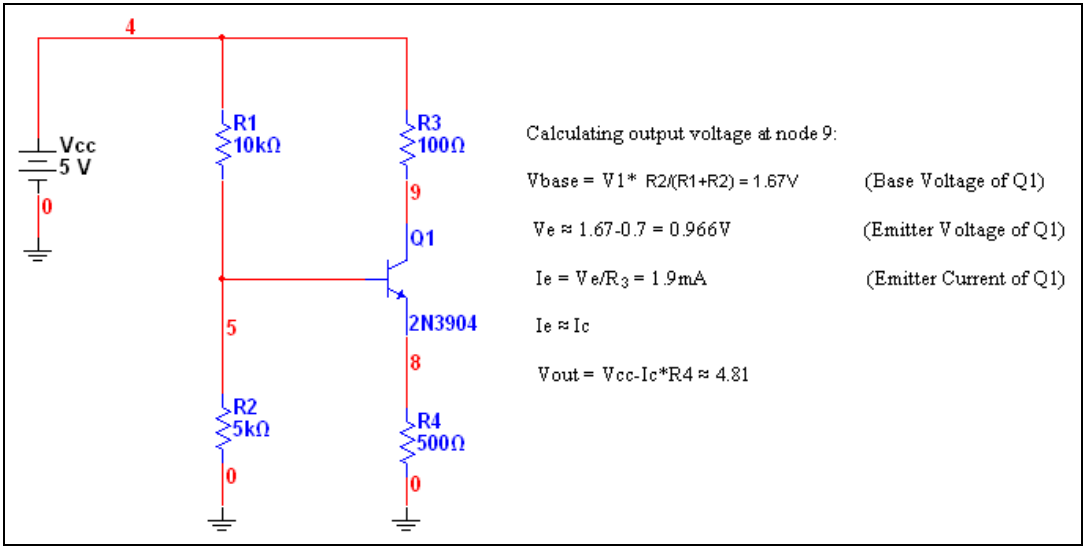
Refer to the *Working with Analyses* section for details on the other tabs in the analysis window.

Sample Circuit

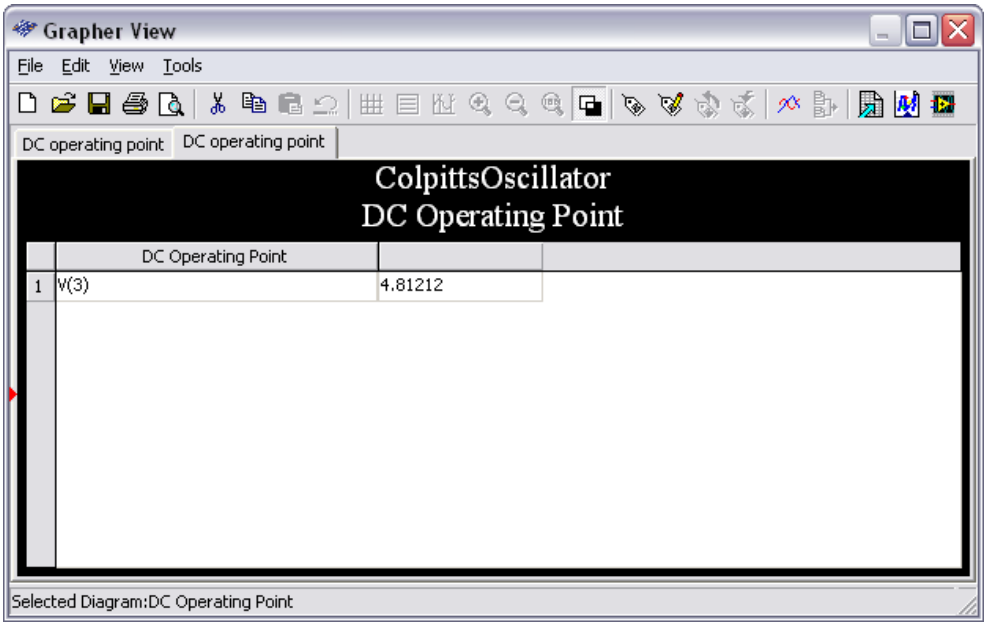
This example analyses the DC Operating Point results at Node #3.



The circuit above is a Colpitts oscillator. When running DC Operating Point Analysis, Multisim reduces the circuit to that shown below:



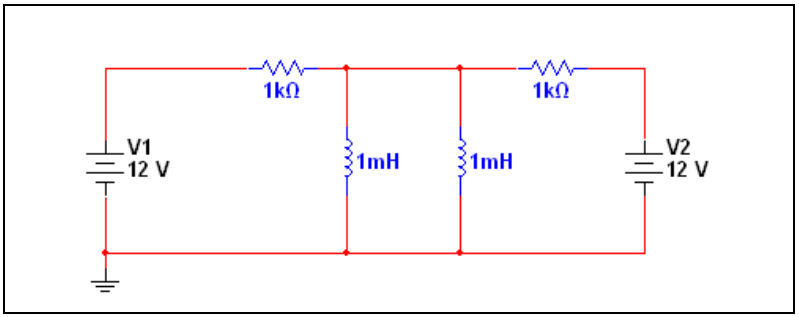
The result of running DC operating point from the sample circuit are shown below. Note that the results match the calculations made in the diagram above.



Troubleshooting DC Operating Point Analysis Failures

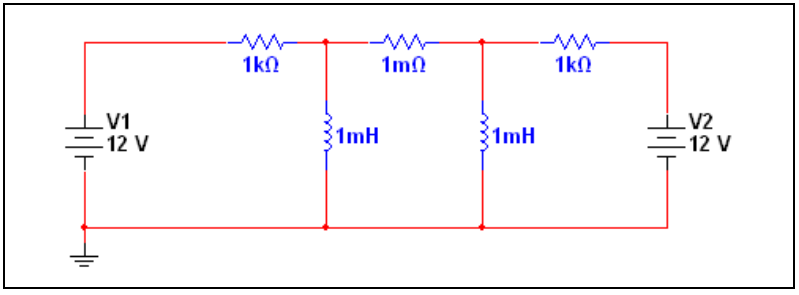
DC Operating Point Analysis may fail to converge for various reasons. The initial estimates for the node voltages may be too far off, the circuit may be unstable or bi-stable (there may be more than one solution to the equations), there may be discontinuities in the models or the circuit may contain unrealistic impedances.

Circuit Failure Example



When current flows through an inductor, voltage develops across the terminals. SPICE models inductors as time-varying current sources. When a circuit has two inductors in parallel, the simulator engine automatically produces an error. When running any analysis for the circuit above, the two inductors are seen as voltage sources by the simulation engine. As a result, this circuit will fail.

To correct this problem, place a 0.001ohm resistor in the circuit. The sources are no longer in parallel, and the DC operating point can converge.



Trouble-shooting Techniques

Use the following techniques to solve many convergence problems and analysis failures. Before you proceed, identify which analysis is causing the problem (keep in mind that **DC Operating Point Analysis** is often performed as the first step of other analyses). In each of the following solutions, begin with step 1, then continue performing the subsequent steps, in order, until the problem is solved.

1. Check the circuit topology and connectivity. Make sure that:
   - the circuit is grounded.
   - the circuit is correctly wired, and includes no dangling nets or stray parts.
   - you haven't confused zeroes with the letter O.
   - the circuit has a ground node and every node in the circuit has a DC path to ground. Make sure no sections of your circuit are completely isolated from ground by transformers, capacitors, etc.
   - Capacitors and voltage sources are not in parallel.
   - Inductors and current sources are not in series.
   - All devices and sources are set to their proper values.
   - All dependent source gains are correct.
   - Your models/subcircuits have been correctly entered.
2. Show all net names. Check the net name assigned to ground components. All grounds MUST be indicated by net name 0 (ZERO). If otherwise, delete a ground and replace by another one from the parts bin.
3. Check for duplicate net names in the circuit. Each net must have a unique name. Reassign if necessary by double-clicking the wire indicating a duplicate net name and typing in another unique name.
4. If working with digital circuits make sure that both, earth and digital grounds are left on the workspace.
5. Copy and then paste your circuit into a new file. Simulate the circuit again.

If the problem persists, adjust the following parameters:

1. From the **Analysis Options** tab of the **DC Operating Point** dialog box, enable **Use Custom Settings** and click **Customize**. The **Analysis Options** dialog box appears.

2. In the **Global** tab:
   - Reduce the RSHUNT value by a factor of 100
   - Increase the GMIN minimum conductance by a factor of 10. Note: GMIN = 1/Rp, where Rp is the smallest parasitic resistance value in the circuit.
3. In the **DC** tab:
   - Set ITL1 = 500 or more. By increasing ITL1, the extra iterations will only be used if they are needed. ITL1 set to 1000 covers about 90% of circuits
   - Set ITL6 = 500
4. Use `.Nodeset` to set a DC voltage level if possible (double-click wire connected to a net in question. Check **Use NODESET for DC** and type in a DC voltage).

Caution When running **DC Operating Point Analysis** for a circuit that contains ammeters and voltmeters (indicators) and their internal settings (resistance) have been changed from their default values, the simulation results indicated by **DC Operating Point Analysis** will be incorrect. Remove ammeters/voltmeters to correct the problem. The results are correct if no changes were applied to ammeter/voltmeter internal settings.

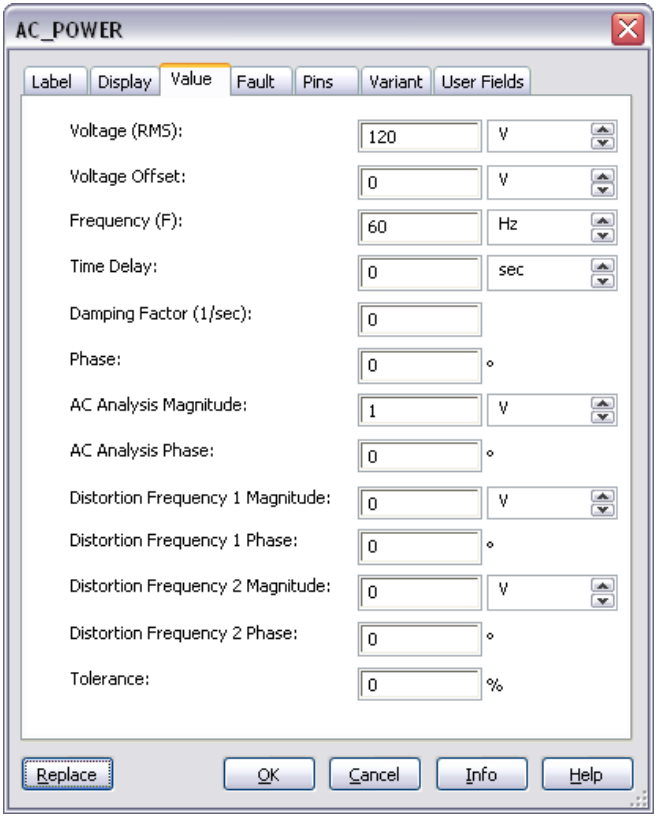
AC Analysis

AC Analysis is used to calculate the frequency response of linear circuits. In **AC Analysis**, the DC operating point is first calculated to obtain linear, small-signal models for all nonlinear components. Then a complex matrix (containing both real and imaginary components) is created. To construct a matrix, DC sources are given zero values. AC sources, capacitors, and inductors are represented by their AC models. Nonlinear components are represented by linear AC small-signal models, derived from the DC operating point solution. All input sources are considered to be sinusoidal. The frequency of the sources is ignored. If the function generator is set to a square or triangular waveform, it will automatically switch internally to a sinusoidal waveform for analysis. **AC Analysis** then calculates the AC circuit response as a function of frequency.

Assumptions—Applied to an analog circuit, small-signal. Digital components are treated as large resistances to ground.

Setting AC Analysis Frequency Parameters

Before you perform the analysis, review your circuit and decide on the nets for analysis. You can specify magnitude and phase of a source for AC frequency analysis by double-clicking on the source and entering the settings in the **Value** tab of the properties dialog box (**AC Analysis Magnitude** and **AC Analysis Phase**) as shown in the figure below. The other settings in this dialog box are used for other analyses or for simulating with the instruments.



Set the following in the **Frequency Parameters** tab of the **AC Analysis** dialog box:

- **Start frequency (FSTART)**—The start frequency for the sweep.
- **Stop frequency (FSTOP)**—The stop frequency for the sweep.
- **Sweep type**—The sweep type: **decade**, **linear**, or **octave**. Defines how points to be calculated are distributed across the frequency range.

- **Number of points per decade**—The number of points to be calculated during the analysis. For a linear sweep type, use the number of points between start and end.
- **Vertical scale**—The vertical scale: **linear**, **logarithmic**, **decimal**, or **octave**. Vertical scale controls the y-axis scaling on the output graph.

Note To reset all parameters to their default values, click **Reset to default**.

The result of the AC frequency analysis is displayed in two parts: gain versus frequency and phase versus frequency.

If you have the Bode plotter connected to your circuit and activate the circuit, a similar analysis is performed.

Setting AC Analysis Frequency Parameters for Normal Use

In most cases, you only need to:

- set a **Start Frequency (FSTART)**
- set a **Stop Frequency (FSTOP)**

Setting AC Analysis Frequency Parameters for Advanced Use

In addition to the frequency range, you can also:

- choose a desired sweep type (decade, linear, or octave) from the **Sweep type** drop-down list
- enter the number of points to be calculated in the **Number of points per decade** field
- choose the vertical scale (linear, logarithmic, decimal or octave) from the **Vertical scale** drop-down list.

Note The greater the number of points calculated, the more accurate the results will be; however, the simulation speed will be adversely affected.

Transient Analysis

In transient analysis, also called time-domain transient analysis, Multisim computes the circuit's response as a function of time. Each input cycle is divided into intervals, and a DC analysis is performed for each time point in the cycle. The solution for the voltage waveform at a node is determined by the value of that voltage at each time point over one complete cycle.

Assumptions—DC sources have constant values; AC sources have time-dependent values. Capacitors and inductors are represented by energy storage models. Numerical integration is used to calculate the quantity of energy transfer over an interval of time.

Setting Transient Analysis Parameters

Before you perform the analysis, review your circuit and decide on the nets for analysis.

Set the **Transient Analysis** parameters using the following:

- **Initial Conditions** drop-down list—Select one of **Zero**, **User-Defined**, **Calculate DC Operating Point**, **Automatically Determine Initial Conditions**.
- **Start time**—Must be greater than or equal to 0 and less than **End time**.
- **End time**—Must be greater than **Start time**.
- **Maximum number of time points** radio button—Click to enter minimum number of time points (number of points between start and stop times).
- **Maximum time step** radio button—Click to enter the maximum time step the simulation can handle.
- **Generate time steps automatically** radio button—Click to generate time steps automatically.

The result of the transient analysis is a calculation of voltage versus time.

If you have the oscilloscope connected to your circuit and activate the circuit, a similar analysis is performed.

Note To re-set all parameters to their default values, click **Reset to default**.

If **Initial Conditions** are set to:

- **Automatically determine initial conditions**—Multisim tries to start the simulation using the DC operating point as the initial condition. If the simulation fails, it uses user-defined initial conditions.
- **Calculate DC operating point**—Multisim first calculates the DC operating point of the circuit, then uses that result as the initial conditions of the transient analysis.
- **Set to zero**—The transient analysis starts from zero initial conditions.
- **User-defined**—The analysis starts from initial conditions as set in the **Transient Analysis** dialog box.

Setting Transient Analysis Parameters for Normal Use

The default settings are appropriate for normal use, providing the transient response of the selected output variables starting at time 0 seconds and stopping after 1 ms (0.001). You can, if you wish:

- change the start time by entering a value greater than or equal to 0 and less than the **End time** in the **Start time (TSTART)** field.
- change the end time by entering a value greater than the **Start time** in the **End time (TSTOP)** field.

Setting Transient Analysis Parameters for Advanced Use

For advanced use, you can:

- define the initial conditions at time 0 seconds by selecting an initial condition (Set to **Zero**, **User-Defined**, **Calculate DC Operating Point**, or **Automatically Determine Initial Conditions**) from the **Initial Conditions** drop-down list.

  You can set the initial conditions to zero, or use the steady state values of the circuit under analysis. During and/or after circuit construction, you can specify node voltages. These forced values can also be used as initial conditions for the analysis.

  If you select "Automatically determine initial conditions", Multisim will attempt to use steady state conditions to run the analysis. If this is unsuccessful, Multisim will set initial conditions to zero. If simulation is still not possible, Multisim will use the specified user-defined conditions.
- define the maximum time step to be taken by the simulation engine by enabling **Maximum time step (TMAX)** and entering the desired time step.
- define the minimum time step to be taken by enabling **Minimum number of time points** and entering the desired number of points to be calculated.
- The value of **TMAX** is determined by dividing the interval between the specified analysis start and end times by the minimum number of time points specified.
- enable **Set Initial Time step (TSTEP)**, and enter a value less than the specified maximum time step value in the **Time step (TSTEP)** field. If possible, the size of the time steps taken during the simulation will begin with the initial timestep and will continue to increase to the value specified by the maximum time step.

Troubleshooting Transient Analysis Failures

If transient analysis is being performed (time is being stepped) and the simulator cannot converge on a solution using the initial time step, the time step is automatically reduced, and the cycle is repeated. If the time step is reduced too far, an error message ("Timestep too small") is generated and the simulation is aborted. If this occurs, try one or more of the following:

- Check the circuit topology and connectivity. See step 1 of *Trouble-shooting Techniques*.
- Set relative error tolerance to 0.01. By increasing the tolerance from 0.001 (0.1% accuracy), fewer iterations are required to converge on a solution and the simulation finishes much more quickly.
- Increase transient time point iterations to 100. This allows the transient analysis to go through more iterations for each time step before giving up.
- Reduce the absolute current tolerance, if current levels allow. Your particular circuit may not require resolutions down to 1 µV or 1 pA. You should allow at least an order of magnitude below the lowest expected voltage or current levels of your circuit.
- Realistically model your circuit. Add realistic parasitics, especially junction capacitances. Use RC snubbers around diodes. Replace device models with subcircuits, especially for RF and power devices.
- If you have a controlled one-shot source in your circuit, increase its rise and fall times.
- Change the integration method to Gear. Gear integration requires longer simulation time, but is generally more stable than the trapezoid method.

Fourier Analysis

Fourier analysis is a method of analyzing complex periodic waveforms. It permits any nonsinusoidal period function to be resolved into sine or cosine waves (possibly an infinite number) and a DC component. This permits further analysis and allows you to determine the effect of combining the waveform with other signals.

Given the mathematical theorem of a Fourier series, the period function f (t) can be written as follows:

$$f(t) = A_0 + A_1 \cos\omega t + A_2 \cos 2\omega t + \ldots + B_1 \sin\omega t + B_2 \sin 2\omega t + \ldots$$

where:

A_0 = the DC component of the original wave

$A_1 \cos\omega t + B_1 \sin\omega t$ = the fundamental component (has the same frequency and period as the original wave)

$A_n \cos n\omega t + B_n \sin n\omega t$ = the n^{th} harmonic of the function

A, B = the coefficients

$\frac{2\pi}{T}$

= the fundamental angular frequency, or 2π times the frequency of the original periodic wave

Each frequency component (or *term*) of the response is produced by the corresponding harmonic of the periodic waveform. Each term is considered a separate source. According to the principle of superposition, the total response is the sum of the responses produced by each term. Note that the amplitude of the harmonics decreases progressively as the order of the harmonics increases. This indicates that comparatively few terms yield a good approximation.

When Multisim performs Discrete Fourier Transform (DFT) calculations, only the second cycle of the fundamental component of a time-domain or transient response (extracted at the output node) is used. The first cycle is discarded for the settling time. The coefficient of each harmonic is calculated from the data gathered in the time domain, from the beginning of the cycle to time point "t". That is set automatically and is a function of the fundamental frequency. This analysis requires a fundamental frequency matching the frequency of the AC source or the lowest common factor of multiple AC sources.

Setting Fourier Analysis Parameters

Before you perform the analysis, review your circuit and select an output node in the dialog box. The output variable is the node from which the analysis extracts the voltage waveform.

Set the **Fourier Analysis** parameters using the following:

- **Frequency resolution (Fundamental frequency)** field—Use to set the frequency of an AC source in your circuit. If you have several AC sources, use the lowest common factor of frequencies. Click the **Estimate** button to have the fundamental frequency estimated.

- **Number of harmonics** field—Use to set the number of harmonics of the fundamental frequency that are calculated.
- **Stop time for sampling (TSTOP)** field—Use to set the amount of time during which sampling should occur.

Note You can also click **Edit transient analysis** to display the **Analysis Parameters** tab of the **Transient Analysis** dialog box. The analysis will use either the **Maximum time step (TMAX)** and **Set initial time step (TSTEP)** values that you enter in this tab, or an automatically calculated minimum value that based on the value in the **Frequency resolution (Fundamental frequency)** field in the **Analysis Parameters** tab of the **Fourier Analysis** dialog box—whichever offers a higher sampling rate (resulting in a more accurate simulation).

Note Refer to the *Setting Transient Analysis Parameters* section for more information.

- **Display phase** checkbox—Enable to display results as phase.
- **Display as bar graph** checkbox—enable to display results as bar graph. If not enabled, results display as linegraph.
- **Normalize graphs** checkbox—Enable to normalize graphs against the 1st harmonic.
- **Display** drop-down list—Choose a display option: **chart**, **graph**, or **chart and graph**.
- **Vertical scale** drop-down list—Choose a vertical scale: **linear**, **logarithmic**, **decibel**, or **octave**.
- **Degree of polynomial for interpolation** checkbox—Enable to enter degree to be used when interpolating between points on simulation
- **Sampling frequency** field—Specify a sampling frequency.

Fourier analysis produces a graph of Fourier voltage component magnitudes and, optionally, phase components versus frequency. By default, the magnitude plot is a bargraph but may be displayed as a line graph.

The analysis also calculates Total Harmonic Distortion (THD) as a percentage. The THD is generated by notching out the fundamental frequency, taking the square root of the sum of the squares of each of the *n* harmonics, and then dividing this number by the magnitude of the notched out fundamental frequency.

$$THD = [(\Sigma_{i=2} V_i^2) / V_1] \times 100\ \%$$

where V_1 is the magnitude of the i^{th} harmonics.

Setting Fourier Analysis Parameters for Normal Use

For normal use, you just need to specify parameters for the following:

- frequency under test, either by clicking **Estimate** to have a value selected based on the AC sources in the circuit, or by entering a value in the **Frequency resolution (Fundamental frequency)** field. This value should be the lowest common factor for the frequencies present in the circuit.
- number of harmonics, by entering a value in the **Number of harmonics** field. You can specify the stopping time for sampling to avoid unwanted transient results prior to the circuit reaching steady-state operation.
- stopping time for sampling by enabling **Stop time for sampling (TSTOP)** and entering a new stopping time for sampling. Although the Nyquist rate specifies only two times the highest frequency component being considered in the analysis as a suitable sampling rate, it is recommended that you specify a sampling frequency sufficient to obtain a minimum of 10 sampling points per period.
- enter a value in the **Sampling frequency** field.

Note The sampling frequency should be equal to the frequency resolution (the number of harmonics plus one) multiplied by at least 10.

Setting Fourier Analysis Parameters for Advanced Use

In addition to the basic procedures, you can also specify parameters for the following:

- degree of polynomial for interpolation, by enabling **Degree of polynomial for interpolation** and entering a value in the appropriate field. The higher the degree of polynomial the greater the accuracy of the results.
- results display format by doing one or all of the following:
  - choosing a vertical scale (linear, logarithmic, decibel or octave) from the **Vertical scale** list.
  - choosing a display option (chart, graph, or chart and graph) from the **Display** list.
  - enabling **Display phase** to display results as phase.
  - enabling **Display as bar graph** to display the results as a bar graph instead of a linegraph.
  - enabling **Normalize graphs** to normalize the results with respect to the frequency under test.

- transient analysis options by clicking **Edit transient analysis** to edit the transient analysis setup. Refer to the *Setting Transient Analysis Parameters for Normal Use* section for more information.

Noise Analysis

Noise is electrical or electromagnetic energy that reduces the quality of a signal. Noise affects digital, analog and all communications systems. Multisim creates a noise model of the circuit, using noise models of each resistor and semiconductor device, instead of AC models and then performs AC-like analysis. It calculates the noise contribution of each component and propogates it to the output of the circuit sweeping through the frequency range specified in the analysis dialog box.

Noise analysis calculates the noise contribution from each resistor and semiconductor device at the specified output node. Each resistor and semiconductor device is considered a noise generator. Each noise generator's contribution is calculated and propogated by the appropriate transfer function to the output of the circuit. The "total output noise" at the output node is the RMS (Root Mean Square) sum of the individual noise contribution. The result is then divided by the gain from input source to the output source to get the "equivalent input noise". This is the amount of noise which, if injected at the input source into a noiseless circuit, would cause the previously calculated amount of noise at the output. The "total output noise" voltage can be referenced to ground or it may be referenced to another node in the circuit. In this case the total output noise is taken across these two nodes.

Multisim can model three different kinds of noise:

1. ***Thermal noise*** (also known as *Johnson*, or *white* noise) is temperature dependent and caused by the thermal interaction between free electrons and vibrating ions in a conductor. Its frequency content is spread equally throughout the spectrum.

   The power of this generated noise is given by Johnson's formula:

   $P = k \; x \; T \; x \; BW$

   where

   k = Boltzmann's constant ($.38 \times 10^{-23}$ *J/K*)

   T = resistor temperature in Kelvin (T = 273 + temperature in Celsius)

   BW = frequency bandwidth of the system being considered

The thermal voltage could be represented by a mean-square voltage source in series with the resistor

$V^2 = 4kTR \times BW$

or the resistor mean-square current generator

$i^2 = 4kTBW/R$.

2. ***Shot noise*** is caused by the discrete-particle nature of the current carriers in all forms of semiconductors. It is the major cause of transistor noise. The equation for shot noise in a diode is:

   $i = (2q \times Idc \times BW)^{1/2}$

   where

   i = shot noise (RMS amperes)

   q = electron charge (1.6×10^{-19} Coulombs)

   Idc = DC current (A)

   BW = bandwidth (Hz)

   For all other devices, such as transistors, no valid formula is available. See the device manufacturer's data sheet. Shot noise and thermal noise are additive.

3. ***Flicker noise*** is usually generated by BJTs and FETs and occurs in frequencies below 1 KHz. This is type of noise is also known as excess noise or pink noise. It is inversely proportional to frequency and directly proportional to temperature and DC current levels.

   $V^2 = k^* Idc/f$

   A component's noise contribution is determined by its SPICE model. Within the model two parameters will affect the output of the noise analysis:

   - AF = Flicker noise component (AF= 0)
   - KF = Flicker Noise (KF=1).

Setting Noise Analysis Parameters

Before you perform the analysis, review your circuit and decide on an input noise reference source, output node and reference node.

Set the **Noise Analysis** parameters using the following:

- **Input noise reference source** drop-down list—Choose an AC voltage source as input.
- **Output node** drop-down list—Choose the node at which all noise contributions will be summed.

- **Reference node** drop-down list—Choose the reference voltage.
- **Calculate power spectral density curves** radio button—Select to generate a graph of the power spectral density.
- **Calculate total noise values** radio button—Select to generate a table with total noise data.

Setting Noise Analysis Parameters for Normal Use

Noise analysis performs an AC analysis to determine the noise. Noise analysis produces an output noise spectrum, an input noise spectrum and, optionally, a component contribution spectrum. When the analysis is finished, its results are displayed as a graph of voltage squared, V^2, versus frequency.

In the **Analysis Parameters** tab, specify:

- **Input noise reference source**
- **Output node**
- **Reference node.**

By default, Multisim will only display nodes that are part of the current page.

Complete the following steps to display nodes contained within subcircuits or hierarchical blocks:

1. Click **Change Filter** and select **Display submodules** from the **Filter Nodes** dialog box that appears.

   All three filter options are explained below:

   - **Display internal nodes**—Displays nodes within hierarchical blocks and subcircuits.
   - **Display submodules**—Displays components within semiconductor devices, determined by the SPICE model of the semiconductor device.
   - **Display open pins**—Displays all unconnected nodes of the circuit.

Setting Noise Analysis Frequency Parameters

Set the **Noise Analysis Frequency Parameters** as follows:

- **Start frequency (FSTART)** field—Enter the start frequency for the sweep.

- **Stop frequency (FSTOP)** field—Enter the end frequency for the sweep.
- **Sweep type** drop-down list—Choose one of **decade**, **linear**, **octave**.
- **Number of points per decade** field—For linear, number of frequency points between start and end frequencies. More points result in more accuracy in your graph.
- **Vertical scale** drop-down list—Choose one of **Logarithmic**, **Linear**, **Decibel**, **Octave**.

Setting Noise Analysis Frequency Parameters for Normal Use

The default settings in the **Frequency Parameters** tab are appropriate for most cases. You just need to define a frequency range by typing a value in the **Start Frequency (FSTART)** field and in the **Stop Frequency (FSTOP)** field.

To copy the settings from the current AC analysis to this analysis, click **Reset to main AC values**.

Once the required variables are selected and the frequency range has been defined, you can then run the analysis.

Setting Noise Analysis Frequency Parameters for Advanced Use

In the **Frequency Parameters** tab you can also set:

- sweep type, by choosing the desired sweep type (decade, linear, or octave) from the **Sweep type** drop-down list. The sweep type defines how the points to be calculated are distributed across the frequency range.
- the number of points to be calculated during the analysis, by entering a value in the **Number of points per decade** field.

Note The greater the number of points calculated, the more accurate the results will be, however, the simulation speed will be adversely affected.

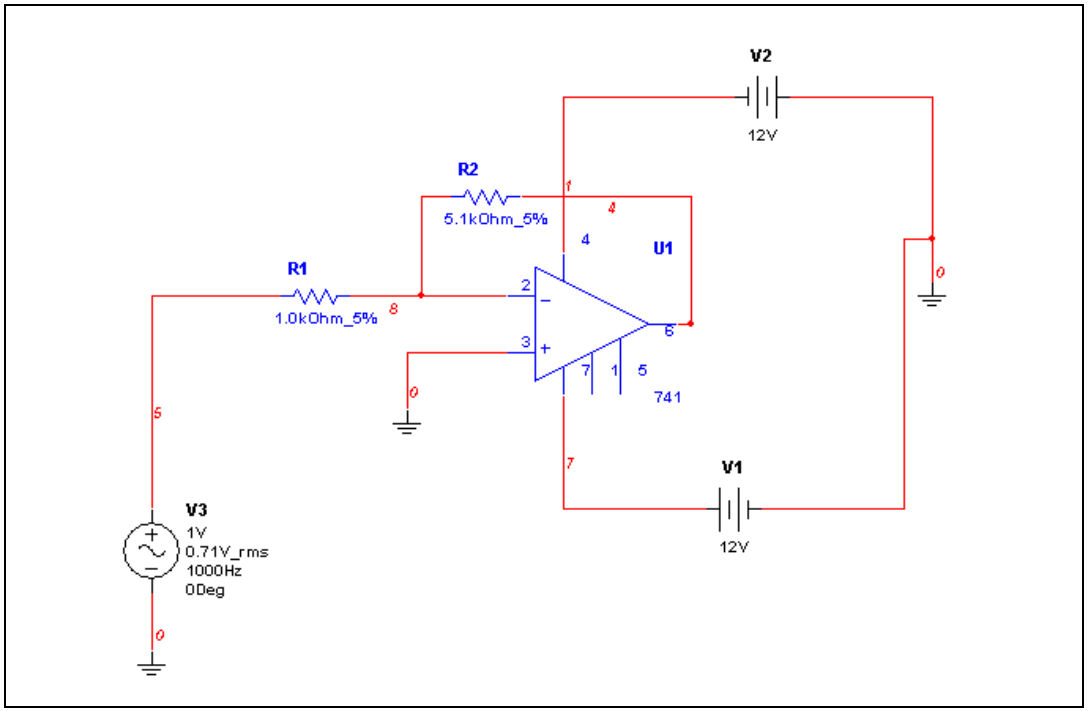
- the format of the analysis results by choosing the desired scale (linear, logarithmic, decimal, or octave) from the **Vertical scale** drop-down list.

Note Click **Reset to Default** to reset all parameters in the **Frequency Parameters** tab to their default values.

Refer to the *Working with Analyses* section for information about othe tabs in the **Noise Analysis** dialog box.

Noise Analysis Example

The circuit below is a basic operational amplifier with a gain of 5. For this analysis, we will obtain the results for the noise voltage for R1 and R2 and display the graph of the noise spectrum across a frequency range between 10 Hz and 10 GHz.



Expected Results

The calculations below will determine what kind of output we are to expect:

If R1=1Kohm, then:

$$noise = 4rkTB$$
$$= 4(1000\Omega)(1.38*10^{-23})(295K)(10^{9}\text{Hz})$$
$$= 162\text{nV}^2$$

If R2=5kOhm then

$$noise = 4rkTB$$
$$= 4(5000\Omega)(1.38*10^{-23})(295K)(10^{9}\text{Hz})$$
$$= 814\text{nV}^2$$

Multisim Analysis

Complete the following steps to set up the analysis in Multisim:

1. Select **Simulate»Analyses»Noise Analysis**.
2. Select the **Analysis Parameters** tab and set the following:
   - **Input noise reference source**—vv3
   - **Output node**—V(4)
   - **Reference node**—V(0)
3. Select the **Frequency Parameters** tab, and set the following:
   - **FSTART**—1Hz
   - **FSTOP**—10GHz
   - **Sweep type**—Decade
   - **Number of points per decade**—5
   - **Vertical Scale**—Logarithmic
4. Select the **Output** tab, select the following variables for plot during simulation:
   - innoise_total_rr1 & innoise_total_rr2

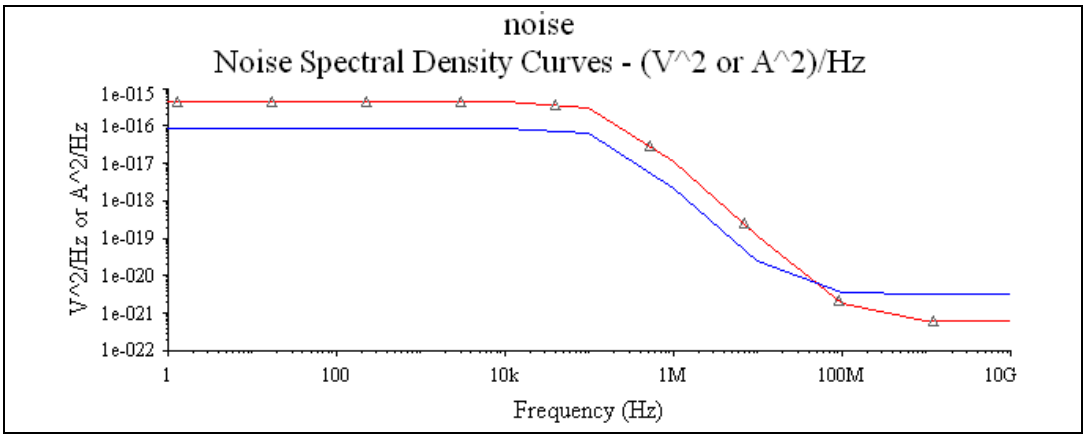
Note It is not necessary to add any devices/model parameters to our list of variables for this example.

5. Click **Simulate**. A chart displays in the **Grapher** with data similar to the expected results.

To view the traces, you must complete the following steps to re-initiate the analysis:

1. Select **Simulate»Analyses»Noise Analysis**.
2. In the **Analysis Parameters** tab, enable **Set points per summary** and enter 5 in the field.
3. In the **Output** tab select the following variables to plot during simulation:
   - onoise_rr1 & onnoise_rr2

4. Click **Simulate**. The **Grapher** displays as follows:



This graph shows that the noise voltage is constant for lower frequencies. For higher frequencies the noise voltage drops considerably.

Distortion Analysis

A perfectly linear amplifier will amplify the input signal without any signal distortion at the output. There are always spurious signal components, however, that are added to a signal in the form of harmonics or intermodulation distortion.

Distortion Analysis is used to analyze signal distortion that may not be evident using transient analysis. Signal distortion is usually the result of gain nonlinearity or phase nonuniformity in a circuit. Multisim simulates harmonic distortion and intermodulation distortion for analog small-signal circuits.

Harmonic Distortion

A perfect linear amplifier can be described using the following formula:

$Y = AX$, where Y is the output signal, X is the input signal and A is the amplifier gain.

The general expression including higher order terms is given by the following:

$Y = AX + BX^2 + CX^3 + DX^4 + \ldots$, where B and C, etc. are the constant coefficients for the higher order terms.

The second term in the above equation is known as the second-order component, the third term is the third-order component and so on. Harmonic distortion can be analyzed by applying a spectrally pure signal source to a circuit design. By analyzing the output signal and its harmonics the distortion can be determined. Multisim will calculate the node voltages and branch currents at the harmonic frequencies $2f$ and $3f$ and display the results against the input frequency f as it is swept across the user defined frequency range.

Intermodulation Distortion

Intermodulation distortion occurs when two or more signals are input into an amplifier simultaneously. In this case, the interaction of the signals produces an intermodulation effect. This analysis will determine node voltages and branch currents at the intermodulation product frequencies of $f_1 + f_2$, $f_1 - f_2$ and $2f_1 - f_2$ vs. the user defined swept frequency.

Multisim Approach

Multisim simulates both harmonic distortion and intermodulation (IM) distortion for analog small-signal circuits. For each AC Source in the circuit you set the parameters to be used with these distortion analyses. Multisim will determine the node voltages and branch currents at each point in the circuit. For harmonic distortion values are determined for the second and third harmonics. For intermodulation distortion the analysis will calculate values at frequencies $f_1 + f_2$, $f_1 - f_2$ and $2f_1 - f_2$.

Preparing the Circuit for Distortion Analysis

Before you perform the analysis you must decide which sources you will use. The distortion analysis parameters are set for each source independently. Follow the steps below for each AC Source that you would like to use in the distortion analysis. To perform harmonic distortion analysis use only Step 1 and 2 below. If you want to perform an intermodulation distortion analysis follow all three steps.

Complete the following steps to set source options for distortion analysis:

1. Double-click on the source.
2. In the **Value** tab set the **Distortion Frequency 1 Magnitude** and **Phase**.
3. In the **Value** tab set the **Distortion Frequency 2 Magnitude** and **Phase**. Only use this setting if you want to perform an intermodulation distortion analysis.

Understanding the Distortion Analysis Options

To start **Distortion Analysis**, select **Simulate»Analyses»Distortion Analysis**.

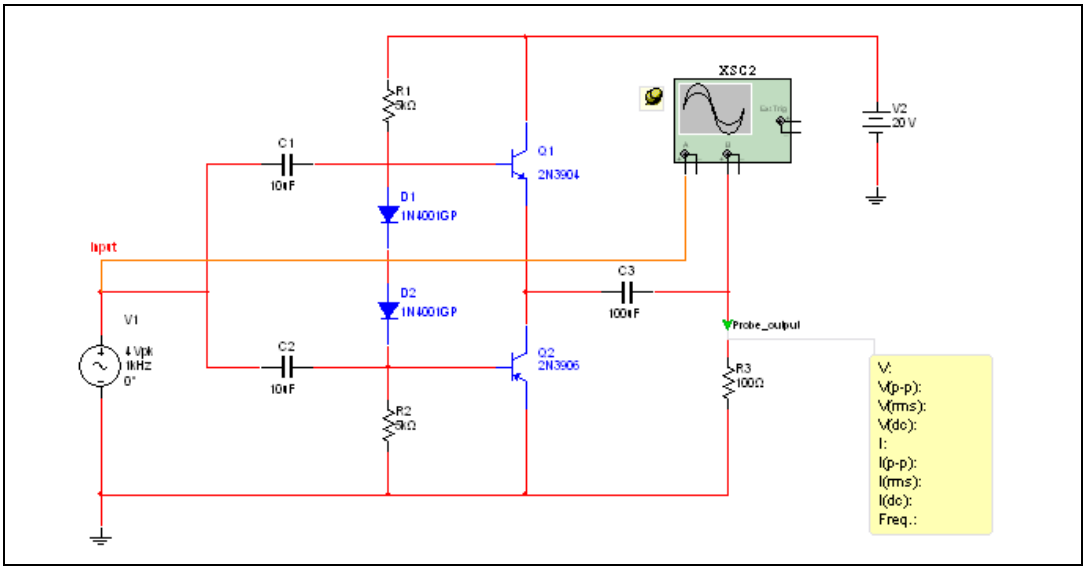
Before you perform the analysis, review your circuit and decide on one or more sources and one or more nodes for analysis.

Set the **Distortion Analysis** parameters as follows:

- **Start frequency (FSTART)** field—Enter the start frequency for the sweep.
- **Stop frequency (FSTOP)** field—Enter the stop frequency for the sweep.
- **Sweep type** drop-down list—Select one of **decade**, **octave**, **linear**.
- **Number of points per decade** field—Enter number of points (for accuracy).
- **Vertical scale** drop-down list—Units for vertical axis.
- **F2/F1 ratio** checkbox—For intermodulation distortion only. When enabled, if there are signals of two frequencies (F1 and F2), then F2 is set to this ratio multiplied by the start frequency while F1 sweeps. Must be greater than 0.0 and less than 1.0.

Distortion Analysis for Harmonic Distortion

To illustrate the settings and behavior of this analysis, we will use the following circuit. This is an example of a Class B Push-Pull Amplifier.

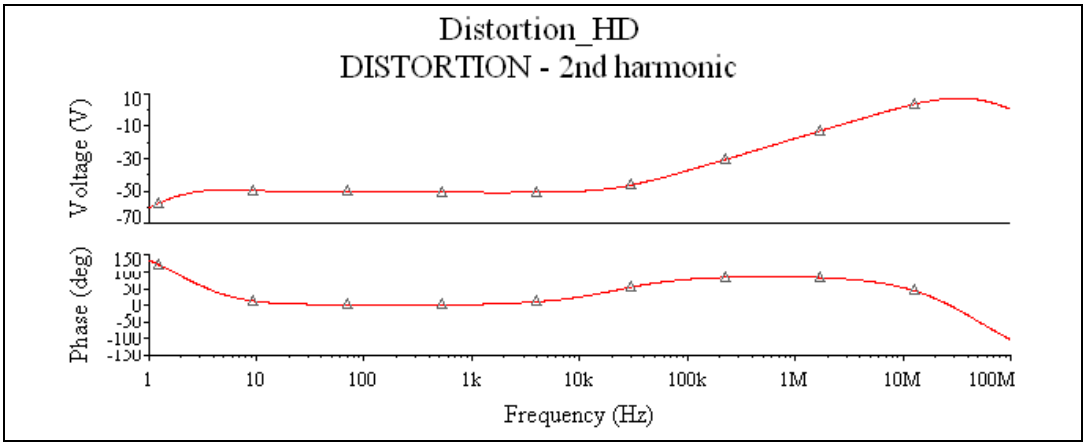


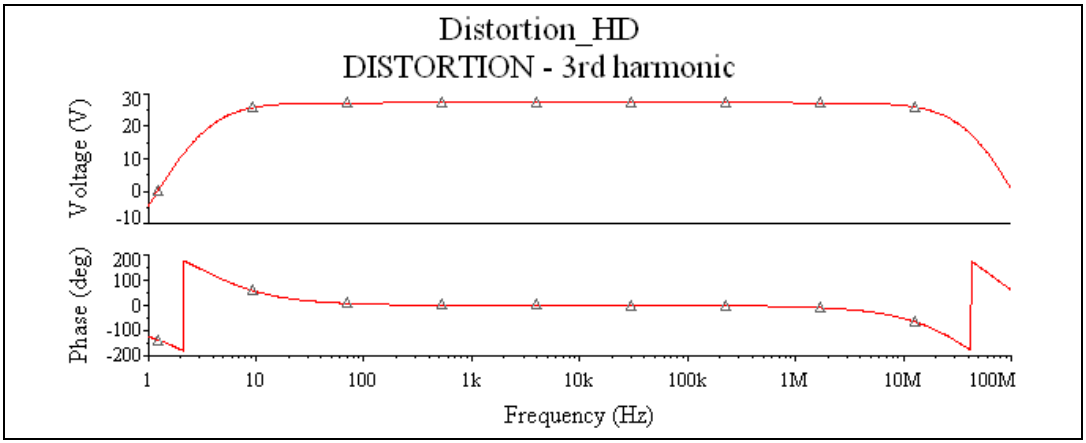
Complete the following steps to set up **Distortion Analysis** (to analyze Harmonic Distortion):

1. Double-click on the AC Source and select the **Value** tab:
   - Choose **Distortion Frequency 1 Magnitude** and set the input amplitude to 4 V and leave the phase at zero.
   - Click **OK**.
2. Select **Simulate»Analyses»Distortion Analysis**.
3. Select the **Analysis Parameters** tab and set the following:
   - Set the **Start frequency (FSTART)** to 1 Hz
   - Set the **Stop frequency (FSTOP)** to 100 MHz
   - Set the **Sweep type** to Decade
   - Set the **Number of points per decade** to 100
   - Set the **Vertical scale** to Decibel

Note Leave **F2/F1 ratio** unchecked.

4. Select the **Output** tab.
   - Select the variable “V(output)” from the list **Variables in circuit**.
   - Click **Add**. The variable “output” will move to the **Selected variables for analysis** list.
5. Click **Simulate**. Two graphs will be displayed with the second harmonic distortion results on one graph and the third harmonic distortion results on the other.





Distortion Analysis for Intermodulation Distortion

This example also uses the Class B Push-Pull Amplifier from above.

Complete the following steps to set up **Distortion Analysis** (to analyze Intermodulation Distortion):

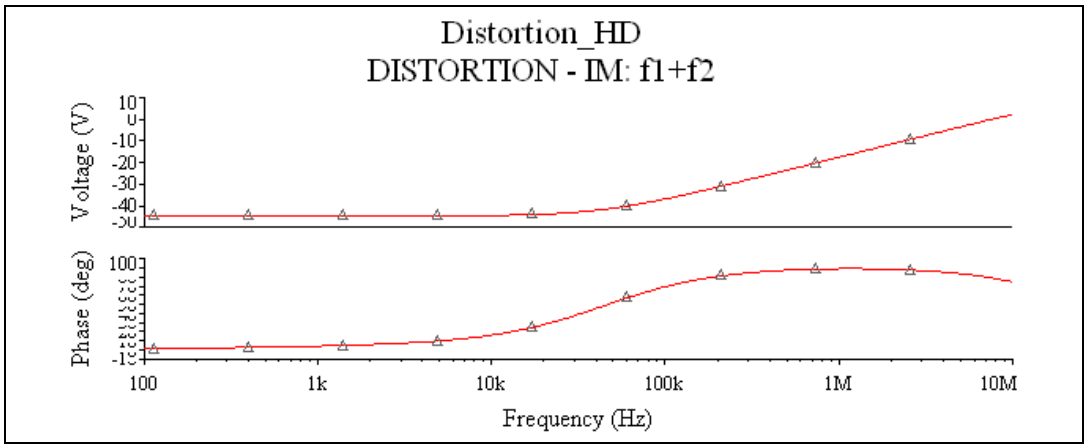
1. Double-click on the AC Source and select the **Value** tab:
   - Choose **Distortion Frequency 1 Magnitude** and set the input amplitude to 4 V and leave the phase at zero.
   - Choose **Distortion Frequency 2 Magnitude** and set the input amplitude to 4 V and leave the phase at zero.
   - Click **OK**.
2. Select **Simulate»Analyses»Distortion Analysis**.
3. Select the **Analysis Parameters** tab and set the following:
   - Set the **Start frequency (FSTART)** to 100 Hz.
   - Set the **Stop frequency (FSTOP)** to 10 MHz
   - Set the **Sweep type** to Decade
   - Set the **Number of points per decade** to 100
   - Set the **Vertical scale** to Decibel
   - Check the **F2/F1 ratio** and set the value to 0.499997. As F1 is swept the analysis frequency will change according to this product of the ratio and the initial value of F1 (FSTART.)

Note The ratio F2/F1 must be greater than 0 and less than 1. Also, this number should be an irrational number but because of the finite precision of the computer this is not possible. It is best, therefore, to use a floating point number with a large number of digits.

4. Select the **Output** tab.
   - Select the variable "V(output)" from the list **Variables in circuit**.
   - Click **Add**. The variable "V(output)" moves to the **Selected variables for analysis** list.
5. Click **Simulate**. Three graphs will be displayed with plots for frequencies F1 + F2,
   F1 – F2 and 2F1 – F2.

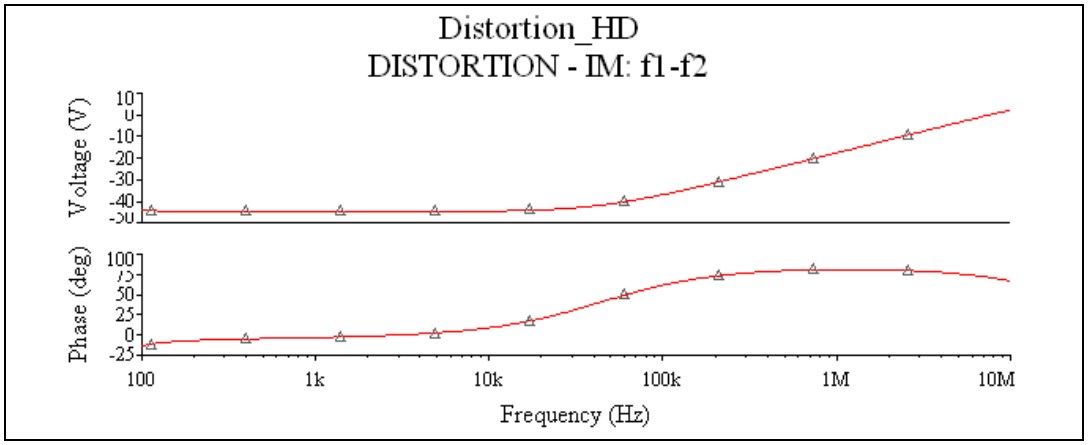
Note F2 is the product of the ratio (F2/F1 ratio) and the initial value of F1 (FSTART.)

The first graph plots the F1+F2 harmonic.



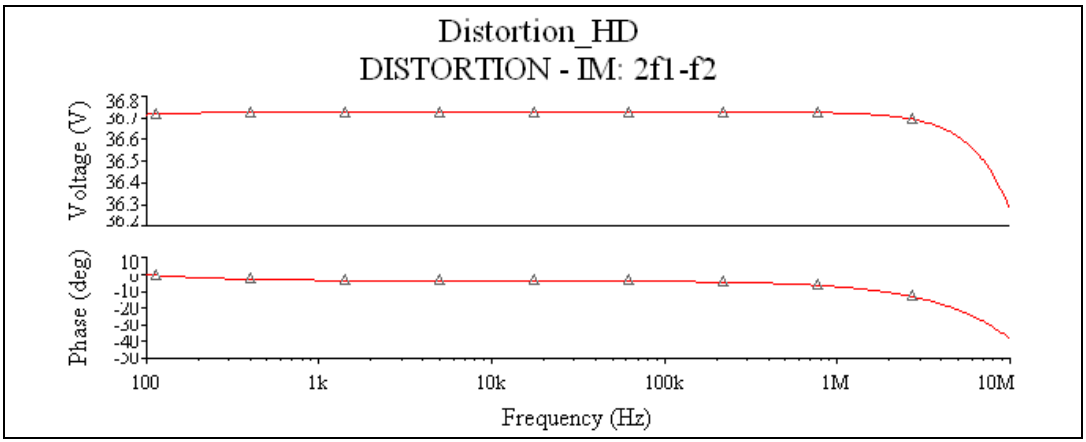
Notice as F1 reaches higher frequencies (100K to 10M), if mixed with F2 (50Hz), the F1+F2 harmonic will increase in magnitude dramatically. At these frequencies, filtering may be needed (if possible) to separate the F1+F2 harmonic from the input signal (at the circuit output).

The second graph plots the F1-F2 harmonic.



This harmonic is very similar in response to the F1+F2 harmonic. The same analysis would apply.

The third graph plots 2F1-F2 harmonic.



From the third graph, we notice that the 2F1-F2 harmonic is higher in magnitude than the first two harmonics (F1+F2 and F1-F2). Although the magnitude of this harmonic is low, filtering this component may be necessary.

DC Sweep Analysis

As this analysis is performed in Multisim, the following procedure is performed.

1. The DC Operating Point is taken.
2. The value from the source is incremented and another DC Operating Point is calculated.

This procedure allows you to simulate the circuit many times, sweeping the DC values within a pre-determined range. You can control the source values by choosing the start and stop values and the increment for the DC range. The bias point of the circuit is calculated for each value of the sweep. To calculate the DC response of the circuit, SPICE treats all capacitors as open circuits, all inductors as shorts, and uses only DC values of voltage and current sources.

Setting DC Sweep Analysis Parameters

Before setting analysis parameters, review your circuit and decide on one or two DC sources to sweep, and a node for analysis.

DC Sweep Analysis plots the appropriate curves sequentially. If only one source is being swept, a curve of the output node value versus source value is traced. If two sources are swept, the number of curves equals the number of points for the second source. Each curve represents the output node value versus the first source value while the second source value is held at each of its sweep values.

Set the **DC Sweep Analysis** parameters for **Source 1** and **Souce 2** as required:

- **Source** list—Choose the source for the sweep. Optionally, click **Change Filter** to change the items in the **Source** list.
- **Start value** and **Stop value** fields—The values at which the analysis will stop and start.
- **Increment** field—The values by which each sweep will increase.

Setting DC Sweep Analysis Parameters

For normal use, you only need to set:

- the source for the sweep, by choosing from the **Source** drop-down list under the **Source 1** options
- a starting value for the sweep, by entering it in the **Start Value** field
- a stop value for the sweep, by entering it in the **Stop Value** field
- an increment value for the sweep, by entering it in the **Increment** field.

The analysis will calculate the circuit's bias point values beginning with the specified start value. The **Increment** value will then be added to the start value and the circuit variables will be recalculated. The **Increment** value is added again and the process continues until the stop value is reached.

Setting DC Sweep Analysis Parameters Using Filters

You can filter the variables displayed to include internal nodes (such as nodes inside a BJT model or inside a SPICE subcircuit), open pins, as well as output variables from any sub-modules contained in the circuit.

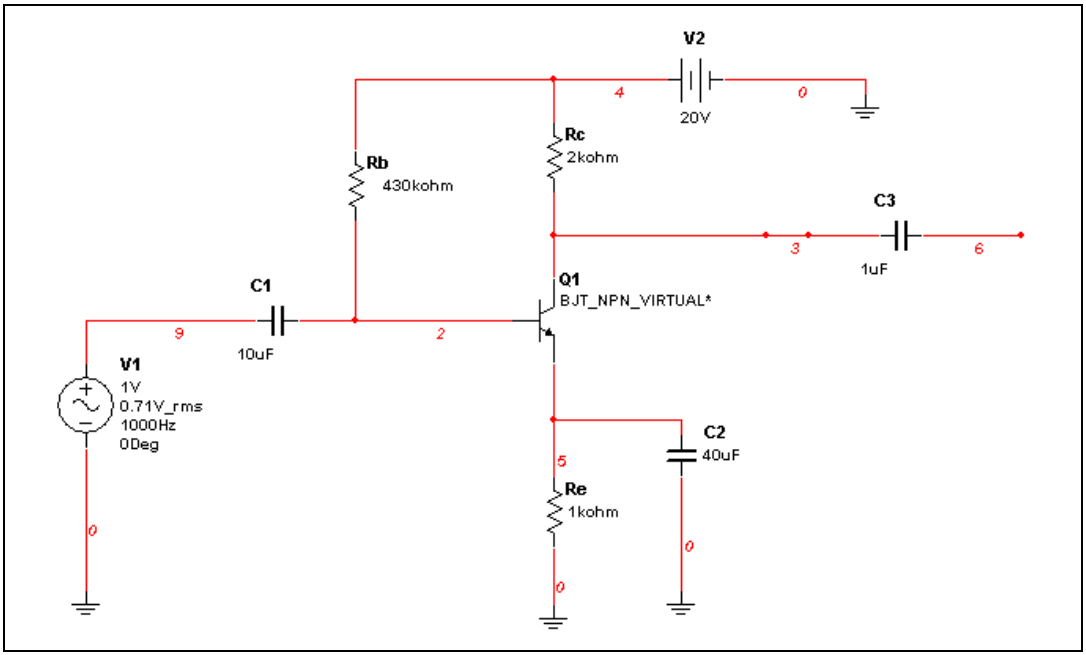
Complete the following steps to filter the variables displayed:

1. Click **Change Filter**. The **Filter Nodes** dialog box appears.
2. Enable one or more settings.
3. Click **OK**.

Refer to the *Working with Analyses* section for details on the other tabs in the analysis window.

DC Sweep Analysis Examples

In this example we will vary V2 from 0 V to 20 V and observe the output at node 3.



Example 1

Expected Results:

$$Ib = \frac{Vcc - Vbe}{Rb + (\beta + 1) * \mathrm{Re}} = \frac{20V - 0.7V}{430k\Omega + (51)(1k\Omega)} = \frac{19.3V}{481k\Omega} = 40.1\mu A$$

$$Ic = \beta * Ib = 50(40.1\mu A) = 2.01mA$$

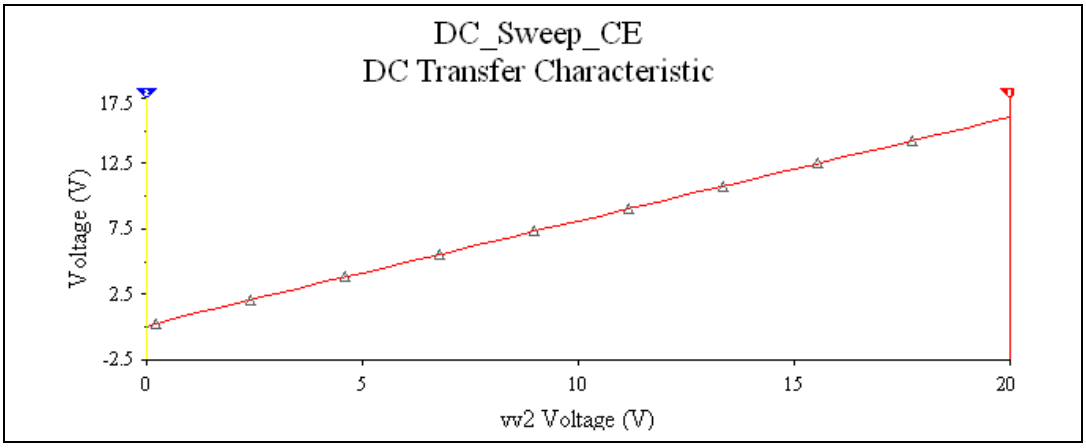
$$Vc = Vcc - IcRc = 15.98\ V$$

where:

Ib = Base Current
β = Gain (= 50)
Vcc = Voltage Source (V1)
Vbe = Voltage from Base to Emitter
Ic = Collector current
Vc = Collector Voltage

Complete the following steps to set up the analysis:

1. Select **Simulate»Analyses»DC Sweep** and click on the **Analysis Parameters** tab.
2. Enter the values below:
   - **Source**—vv2
   - **Start value**—0 V
   - **Stop value**—20 V
   - **Increment**—0.5 V.
3. In the **Output** tab, move **V(3)** to the **Selected variables for analysis** column.
4. Click on the **Simulate** button. The following displays in the grapher:



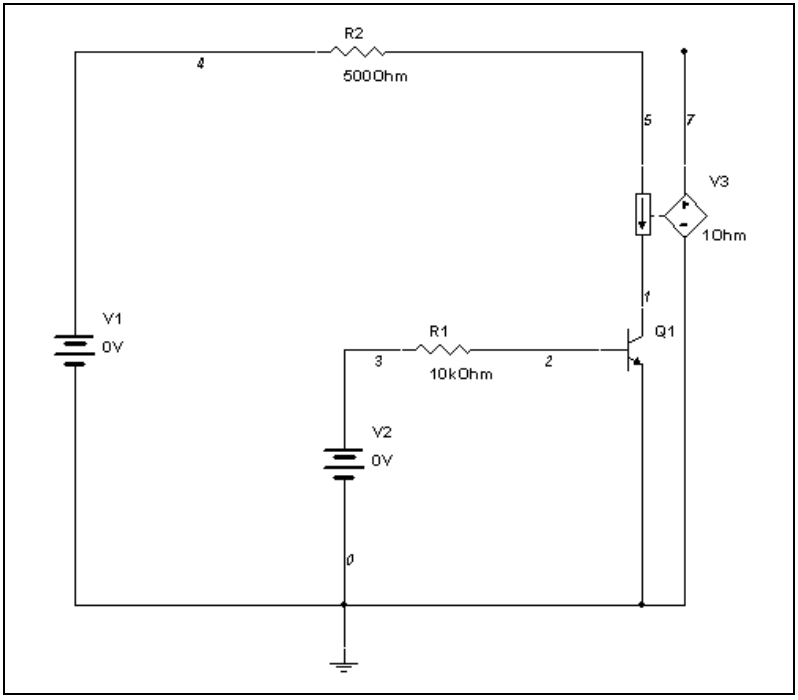
5. To view the cursors and DC transfer characteristics, select in the grapher **View»Show/Hide Cursor**. Two cursors will be displayed. Move cursor 2 to the left side of the graph, and move cursor 1 to the right and line it up along the x-axis with vv2 = 20 V.

   The cursor set at V= 20 V (x1), in the figure above displays the transfer characteristics for y1, which displays the value of 15.9982 V. Therefore the graph displays when the DC source V2 is set to 20 V, then the output form the collector of the transistor will be 15.99 V.

Example 2 (Nested DC Sweeps)

The following example displays the characteristics of the transistor in a common emitter configuration. The graph will display the operating points for this transistor, which will define the region in which amplification will occur.

The current controlled voltage source (V3) has been added so that current flow through the collector can be converted to a voltage for display on the grapher.

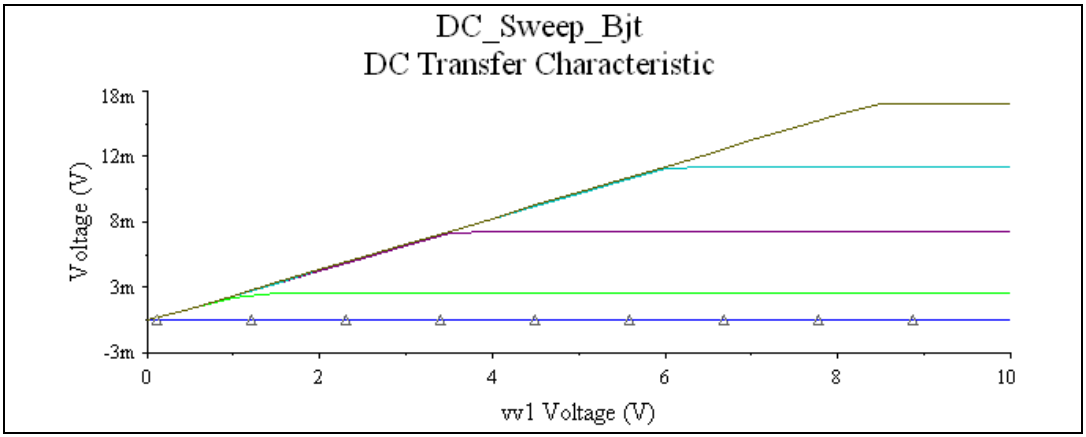


Complete the following steps to set up the analysis:

1. Enter the following in the **Analysis Parameters** tab for **Source 1**:
   - **Source**—vv1
   - **Start value**—0 V
   - **Stop value**—10 V
   - **Increment**—0.5 V.
2. Enter the following in the **Analysis Parameters** tab for **Source 2**:
   - **Source**—vv2
   - **Start value**—0 V
   - **Stop value**—2.5 V
   - **Increment**—0.5 V.

3. In the **Output** tab, move **V(7)** to the **Selected variables for analysis** column. This is the output node at the current controlled voltage source.

4. Click **Simulate**. The following displays in the grapher:



DC and AC Sensitivity Analyses

Sensitivity Analyses help identify how much a circuit component will affect the output signal. As a result, critical components can be specified with tighter tolerances, and are the best candidates for optimization. Likewise, the least critical components can be identified, which make good candidates for cost reduction as their precision does not critically affect design performance.

Sensitivity Analysis calculates the sensitivity of an output node voltage or current with respect to the parameters of a component(s) in your circuit. Sensitivity is expressed as the change in output per unit change of input (both in values and percentages).

Sensitivity Analysis Parameters

Sensitivity Analysis includes:

- DC Sensitivity
- AC Sensitivity.

Both analyses calculate the change produced in an output voltage or current by changing each parameter independently. The results of the DC sensitivity analysis are displayed in tabular form on the grapher.

The AC sensitivity analysis plots the AC graphs for each parameter selected over the user defined frequency range.

When setting up **Sensitivity Analysis**, the following options must be specified:

Output nodes/currents:

- **Voltage**: Sensitivity Analysis will look at node voltages.
  - **Output node** - select the node to examine
  - **Output reference** - select the reference point for the Output node. Usually this is node 0 (Ground).
- **Current**: Output current will be used to determine sensitivity.
- **Expression**: directly enter an output expression for analysis in the **Output Expression** field or click **Edit** to display the **Analysis Expression** dialog box. Refer to the *Adding Analysis Expressions* section for more information.

Output Scaling: Select either **Absolute** or **Relative**:

- **Absolute**—The output voltage or current change per unit of change of the selected component parameter.
- **Relative**—The output voltage or current change per relative change of the selected component parameter relative to the voltage or current when the component parameter is unchanged.

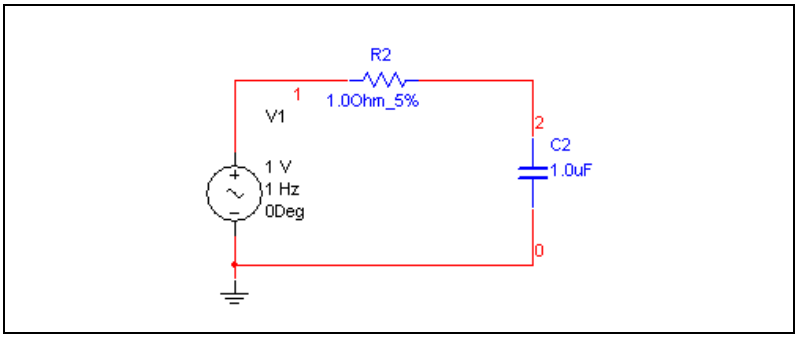
Analysis Type:

- **DC Sensitivity**—DC sensitivity analysis generates a report of the output voltage at a circuit node with respect to all components and their parameters. You can choose to run a DC sensitivity analysis of the current source or voltage source.
- **AC Sensitivity**—AC sensitivity analysis plots an AC graph (frequency domain) for each parameter of the component.

Setting Up and Running Sensitivity Analysis

Example 1

This example determine the AC sensitivity of the circuit shown below to the resistance of R1 at node 2.

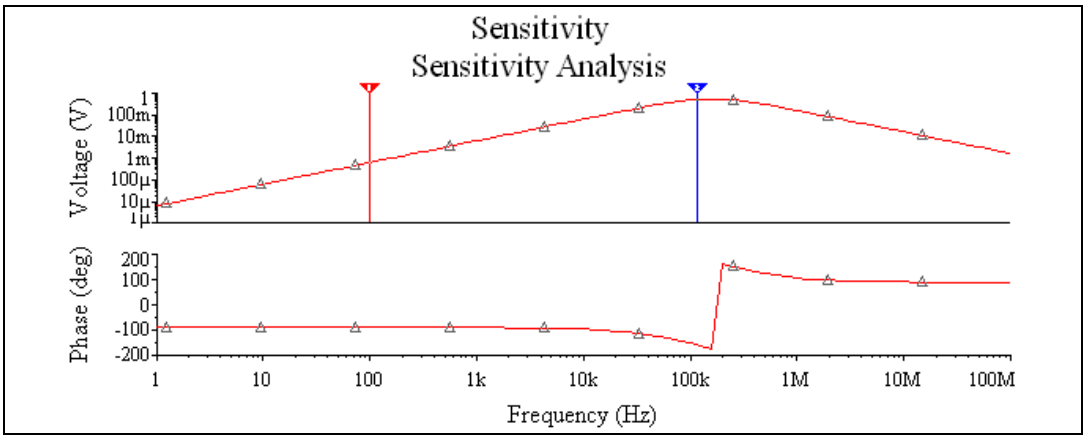


Complete the following steps to set up **Sensitivity Analysis**:

1. Select **Simulate»Analyses»Sensitivity**.
2. Select the **Analysis Parameters** tab and set the following:
   - Set **Output nodes/currents** to **Voltage**.
   - Set **Output node** to V(2) and **Output reference** to V(0).
   - Set **Output scaling** to **Absolute**.
   - Set **Analysis type** to **AC Sensitivity**.
3. Click **Edit Analysis** to display the **Sensitivity AC Analysis** dialog box and set the following parameters:
   - **Start frequency (FSTART)**—1 Hz
   - **Stop frequency (FSTOP)**—100 MHz
   - **Sweep type**—Decade
   - **Number of poits per decade**—10
   - **Vertical scale**—Logarithmic.
4. Select the **Output** tab:
   - Select **rr2** under **Variables in circuit**. You may need to select **Filter Unselected Variables**, select all items and choose **OK** to see the resistor in the selection area.
   - Click **Add**.

   The rr2 variable will move to the **Selected variables for analysis** field.

5. Click **Simulate**. The following is displayed:



The results will plot the output voltage change if R2 is changed by 1Ω from 1Hz to 100MHz.

It is possible to calculate the Sensitivity results manually. However, this can only be done for one frequency value at a time. We will analyze the same circuit at 100Hz.

Using the Sensitivity results shown above the analysis reports that at 100 Hz, the output will change by 628 uV. To calculate this value, we will follow the instructions below:

$$X_C = \frac{1}{2\pi fc}$$

$$X_C = \frac{1}{2\pi * 100 * 1 * 10^{-6}} \qquad \text{(at 100 Hz)}$$

Therefore:

$$X_C = 1.59K$$

$$V_{out} = \frac{1v * 1\Omega}{1\ \Omega - j1.59\ \text{K}} \qquad \text{(voltage at node 2)}$$

$$V_{out} \approx \frac{1}{-j1.59K}$$

$$V_{out} = 628\ \text{uv}$$

If R1 changes by a unit (R1 = 2 Ω),

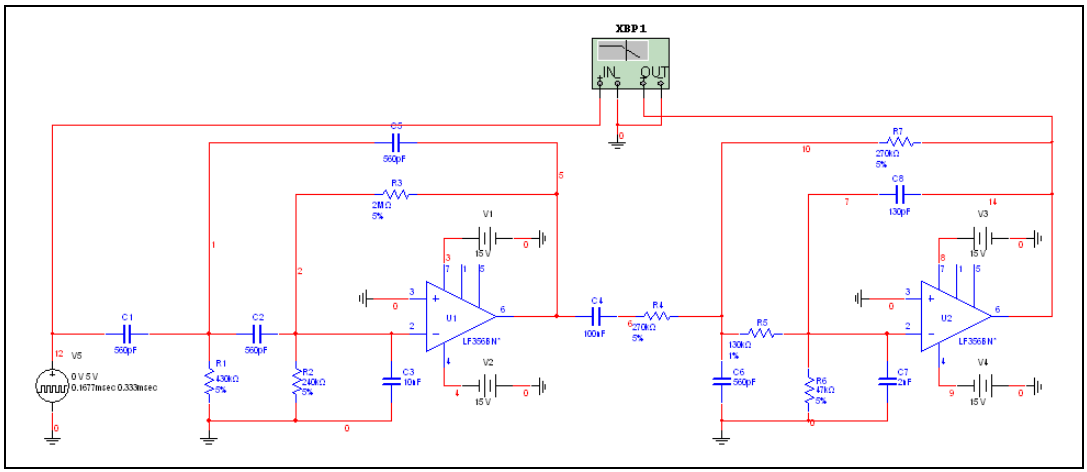
$$V_{out} = \frac{1v * 2\ \Omega}{2\ \Omega - j1.59K}$$

$$V_{out} = 1.2579\ \text{mv}$$

Therefore, the voltage change is 629 uV. This is supported by the results calculated by Multisim.

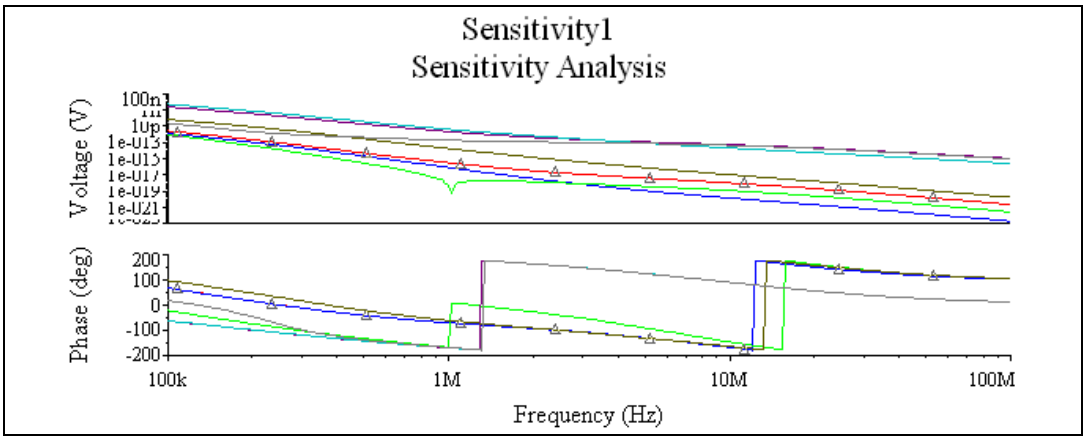
Example 2

To truly understand the power of this analysis, we will analyze the speech filter circuit shown below. When building this circuit, all components will have tolerances. Therefore, the circuit output will always change slightly from simulated results to real results. We will use **Sensitivity Analysis** to predict which component(s) will affect the circuit dramatically if their values are slightly different from expected.



Complete the following steps to start **Sensitivity Analysis**:

1. Select **Simulate»Analyses»Sensitivity**.
2. Select the **Output** tab, set the following:
   - Select rr1 under **Variables in circuit** field.
   - Click **Add**.
   - Repeat for rr2 to rr7.
3. Select the **Analysis Parameters** tab, set the following:
   - Set **Output Nodes/currents** to **Voltage**.
   - Set **Output node** to V(14) and **Output reference** to V(0).
   - Set **Output scaling** to Absolute.
   - Set **Analysis type** to AC Sensitivity.
4. Click **Simulate**. The **Grapher** will plot the sensitivity for all resistors in the design.



From the graph, we can determine that if any resistor is changed by 1 ohm, the results will not change dramatically. However, R7 will cause the biggest change in the circuit output.

Parameter Sweep Analysis

With **Parameter Sweep Analysis**, you can verify the operation of a circuit by simulation across a range of values for a component parameter. The effect is the same as simulating this circuit several times, once for each value. You control the parameter values by choosing a start value, end value, type of sweep that you wish to simulate and the desired increment value in the **Parameter Sweep** dialog box. There are three types of analysis that can be performed on the circuit while the component is manipulated: **DC Operating Point**, **Transient Analysis** and **AC Analysis**.

You will find that some components have more parameters that can be varied than others. This will depend on the model of the component. Active components such as op-amps, transistors, diodes and others will have more parameters available to perform a sweep than passive component such as resisters, inductors and capacitors. For example the inductance is the only parameter available for an inductor as compared to a diode model that contains approximately 15 to 25 parameters.

Setting Parameter Sweep Analysis Parameters

The behavior of a circuit is affected when certain parameters in specific components change. Before you perform the analysis, review your circuit and decide on a component and parameter to sweep, and a node for analysis.

Set the **Paramater Sweep** as follows:

- **Sweep parameter** drop-down list—Choose **Device Parameter** or **Model Parameter**.
- **Device Type** drop-down list—Choose the type of device to sweep.
- **Name** drop-down list—Enter the component to sweep.
- **Parameter** drop-down list—Choose the device parameter of the component to be swept.
- **Sweep Variation Type** drop-down list—Dictates how to calculate the interval between the stop and start values. Choose from **Decade**, **Octave**, **Linear**, or **List**.
- **Value List**—For List sweep only. A list of values to sweep over. Items in the list must be separated by spaces, commas or semicolons.
- **Analysis to sweep** drop-down list—Choose **DC Operating Point**, **AC Analysis**, **Transient Analysis**, or **Nested Sweep**.
- **Group all traces on one plot** checkbox—If not enabled, each trace appears on a separate plot.
- **Edit Analysis** button—Click to edit parameters of the chosen analysis. (For Nested Sweep analysis, see *Nested Sweep Analyses*.)

Parameter sweep analysis plots the appropriate curves sequentially. The number of curves is dependent on the type of sweep as shown below:

- **Linear**—The number of curves is equal to the difference between the start and end values divided by the increment step size.
- **Decade**—The number of curves is equal to the number of times the start value can be multiplied by ten before reaching the end value.
- **Octave**—The number of curves is equal to the number of times the start value can be doubled before reaching the end value.

Setting Parameter Sweep Analysis Parameters

Complete the following steps to set up the analysis parameters:

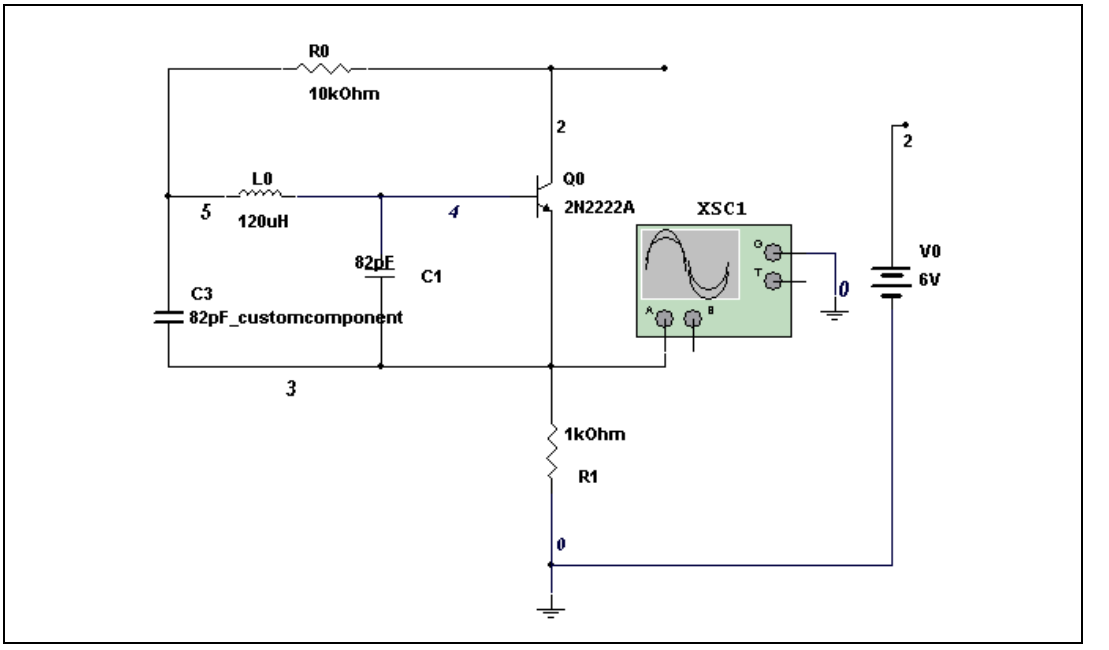
1. Select a sweep parameter by choosing a parameter type (Device or Model) from the **Sweep Parameter** drop-down list, then enter information in the **Device**, **Name**, and **Parameter** fields.

   A brief description of the parameter appears in the **Description** field and the present value of the parameter is displayed in the **Present Value** field.
2. Set the sweep variation type by choosing either linear, decade, or octave from the **Sweep Variation Type** drop-down list.
3. Select the analysis to sweep by choosing from the **Analysis to sweep** drop-down list.
4. If you want to sweep other than the list, type the desired parameter values, separated by a space, in the **Values** field.
5. Optionally, you can set the analysis parameters by clicking **Edit Analysis**. The analysis parameters available depend upon the analysis selected. See the respective analysis elsewhere in this chapter for details on setting these parameters.

Note If the analysis is unedited, the last values set for the analysis will be used. If the analysis has not been run previously, the default values will apply.

Note You can also perform nested sweeps, combining various levels of device/model parameter sweeps. Refer to the *Nested Sweep Analyses* section for more information.

Parameter Sweep Analysis – Example



The above circuit is a Colpitts oscillator. The output of the circuit will generate a square wave. The analysis to be performed on this circuit will change the value of the inductor and simulate the circuit. If you simulate this circuit you will notice that as you decrease the inductance, the frequency of the signal will decrease.

Before we proceed with the analysis we will calculate the expected results. When the inductor L0 varied the frequency at node 3 (the transistor emitter) will vary according to:

$$fr = \frac{1}{2\pi\sqrt{L C_T}}$$

$$C_T = \frac{C_1 C_2}{C_1 + C_2} = \frac{(82pf * 82pf)}{(82pf + 82pf)} = 41p$$

If L0 = 120uH, then the frequency at the output of the circuit will be:

$$fr = \frac{1}{2\pi\sqrt{(120uH)(41pf)}} = 2.2MHz$$

If L0 = 500uH, then the frequency at the output of the circuit will be:

$$fr = \frac{1}{2\pi\sqrt{(500uH)(41pf)}} = 1.1MHz$$

If L0 = 900uH, then the frequency at the output of the circuit will be:

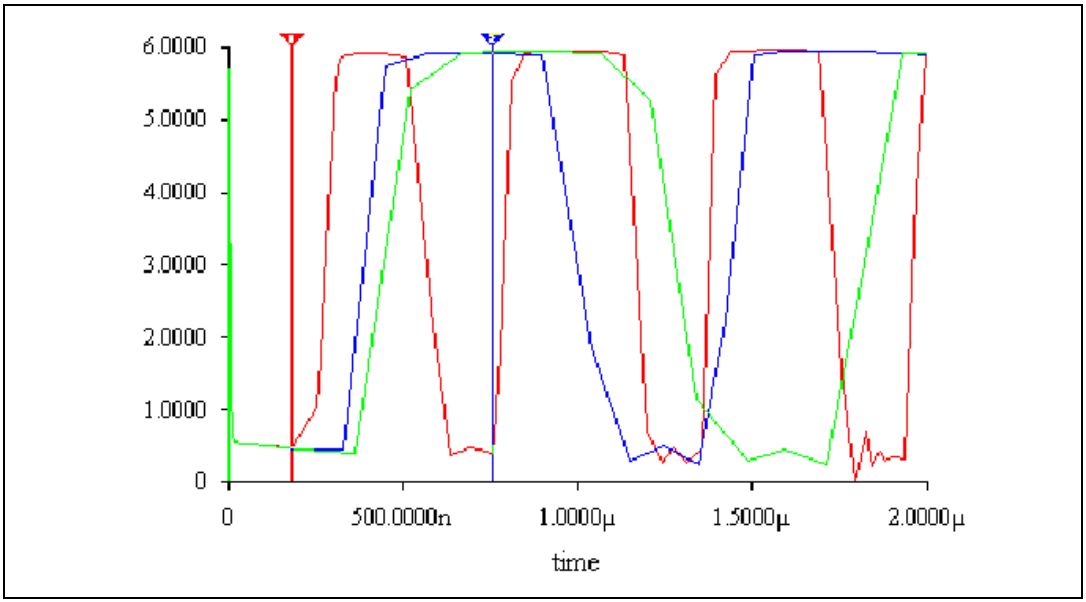
$$fr = \frac{1}{2\pi\sqrt{(900uH)(41pf)}} = 828KHz$$

$$fr = \frac{1}{2\pi\sqrt{(900uH)(41pf)}} = 828KHz$$

Complete the following steps to set up and run the analysis:

1. Select **Simulate»Analyses»Parameter Sweep**.
2. Select the **Analysis Parameters** tab and enter the following parameters:
   - **Sweep parameter** drop-down list—Choose **Device Parameter**.
   - **Device Type** drop-down list—Choose **Inductor**.
   - **Name** drop-down list—Choose **ll0**.
   - **Parameter** drop-down list—Choose **inductance**.
   - **Sweep Variation Type** drop-down list—Choose **List**.
   - **Value List**—Enter **0.00012, 0.0005, 0.0009**.
   - **Analysis to sweep** drop-down list—Choose **Transient Analysis**.
   - **Group all traces on one plot** checkbox—Enable.
3. Click **Edit Analysis** and set the values as follows:
   - **Initial Conditions**— Select **Set to zero**.
   - **Start time (TSTART)—0 Sec**.
   - **End time (TSTOP)—2e-006**.
   - **Maximum time step settings (TMAX)** checkbox—Enable.
   - **Maximum time step (TMAX)** radio button—Enable and enter **1e-005**.
4. Select **OK**.
5. Select the **Output** tab.

6. For this circuit the output variable is at node 3. Under **Variables in circuit** highlight **V(3)** and click on **Add**. Once selected you will notice that variable is under the column **Selected variable for analysis**.
7. Click **Simulate** to generate the results.



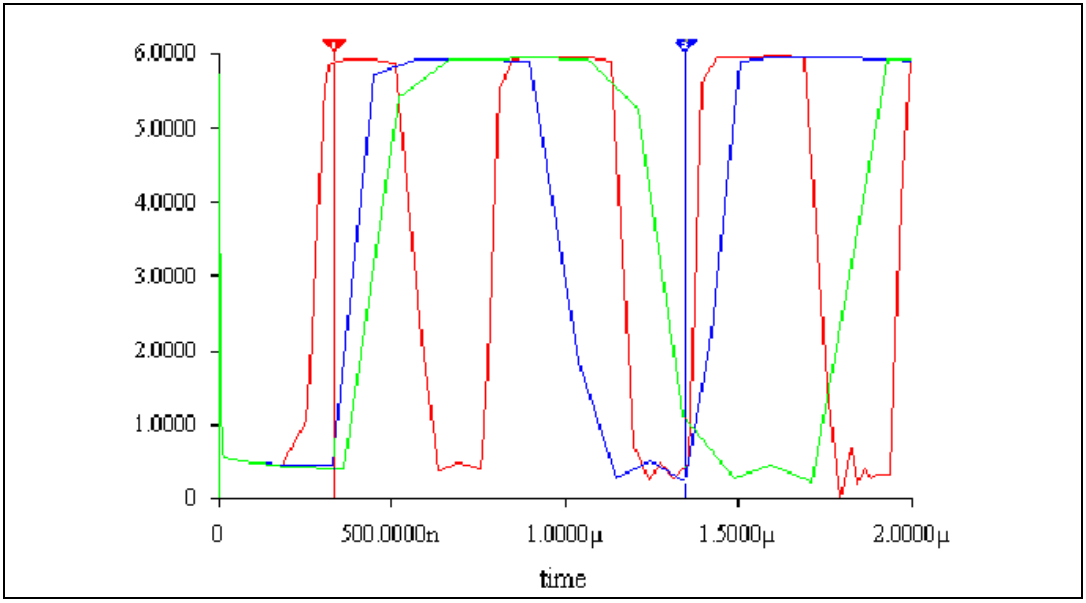
Complete the following steps to view the results:

1. In the graph window select **View»Show/Hide Cursors**.
2. To verify the frequency of the signal when L0 = 120uH, move cursor 1 to the first rising edge of the signal and cursor 2 to the next rising edge as shown in the above graph. To verify the results use the values of X1 and X2 in the **Device Parameter Sweep** table.

$$Time(\text{secs}) = X2 - X1 = 756.05n - 184.93n = 571.12ns$$

$$fr = \frac{1}{Time} = 1.75MHz$$

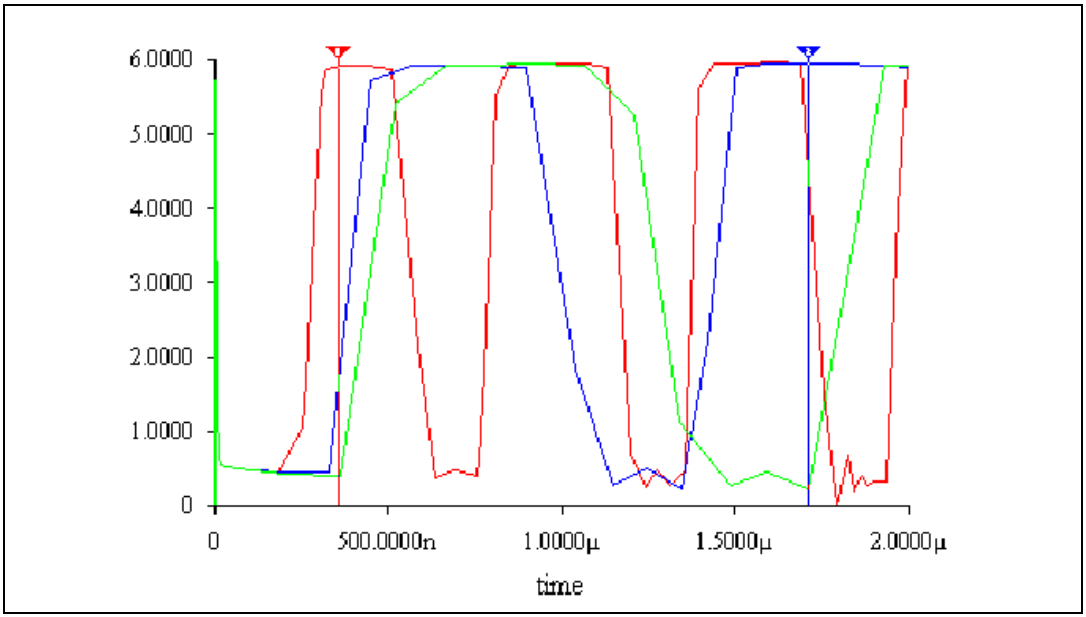
3. To verify the frequency of L0 = 500uH, move cursor 1 to the rising edge of the next signal and cursor 2 to the next rising edge shown below.



$$Time(\text{secs}) = X2 - X1 = 1.3465u - 333.53u = 1.012us$$

$$fr = \frac{1}{Time} = 987.196KHz$$

4. To verify Lo = 900uH move cursor 1 to the rising edge of the next signal and cursor 2 to the next rising edge shown below.



$$Time(\sec s) = X2 - X1 = 1.7129u - 357.299n = 1.355us$$

$$fr = \frac{1}{Time} = 737.680KHz$$

The simulated results are similar to the expected results.

Temperature Sweep Analysis

Using **Temperature Sweep Analysis**, you can quickly verify the operation of your circuit by simulating it at different temperatures. The effect is the same as simulating the circuit several times, once for each temperature. You control the temperature values by choosing start, stop and increment values.

You may perform three types of sweeps: **DC Operating Point**, **Transient Analysis**, and **AC Frequency Analysis**.

Assumptions—See the assumptions for the selected analysis: DC operating point analysis described in *DC Operating Point Analysis*,

transient analysis described in *Transient Analysis*, or AC frequency analysis described in *AC Analysis*.

Temperature sweep analysis affects only components whose model includes temperature dependency, including:

- Virtual Resistor
- 3 - Terminal Depletion N-MOSFET
- 3 - Terminal Depletion P- MOSFET
- 3 - Terminal Enhancement N- MOSFET
- 3 - Terminal Enhancement P- MOSFET
- 4 - Terminal Depletion N- MOSFET
- 4 - Terminal Depletion P- MOSFET
- 4 - Terminal Enhancement N- MOSFET
- 4 - Terminal Enhancement P- MOSFET
- Diode
- LED
- N-Channel JFET
- NPN Transistor
- P-Channel JFET
- PNP Transistor

Setting Temperature Sweep Analysis Parameters

Before you perform the analysis, review your circuit and decide on a node for analysis.

Set the **Temperature Sweep Analysis** parameters as desired:

- **Sweep Parameter** drop-down list—Displays the parameter type.
- **Sweep Variation Type** drop-down list—Dictates how to calculate the interval between the stop and start values. Choose from **Decade**, **Octave**, **Linear**, or **List**.
- **Value List**—For List sweep only. A list of values to sweep over. Items in the list must be separated by spaces, commas or semicolons.
- **Analysis to sweep** drop-down list—Choose **DC Operating Point**, **AC Analysis**, or **Transient Analysis**.
- **Group all traces on one plot** checkbox—If not enabled, each trace appears on a separate plot.
- **Edit Analysis** button—Click to edit parameters of the chosen analysis.

Temperature sweep analysis plots the appropriate curves sequentially. The number of curves is dependent on the type of sweep, as shown below:

- **Linear**—The number of curves is equal to the difference between the start and end values divided by the increment step size.
- **Decade**—The number of curves is equal to the number of times the start value can be multiplied by ten before reaching the end value.
- **Octave**—The number of curves is equal to the number of times the start value can be doubled before reaching the end value.

Setting Temperature Sweep Analysis Parameters

You can use the **Analysis Parameters** tab to define the temperature values to be swept, and the type of analysis to be run at the various swept temperatures. You can also edit the specific analysis being performed.

Complete the following steps to set up and run temperature sweep:

1. Choose the type of distribution (linear, decade, or octave) from the **Sweep Variation Type** drop-down list. To enter a list of temperatures, enter each value separated by a space.
2. Choose the analysis to be performed from the **Analysis to sweep** drop-down list.
3. Click **Edit Analysis** to specify the analysis parameters. The analysis parameters available depend on the analysis selected. See the respective analysis elsewhere in this chapter for details on setting these parameters. If the analysis is unedited, the last values set for the analysis will be used. If the analysis has not been run previously, the default values will apply.
4. Select the **Output** tab and choose the node for which you want to see the results.
5. Click **Simulate** to generate the results.

Transfer Function Analysis

Transfer Function Analysis calculates the DC small-signal transfer function between an input source and two output nodes (for voltage) or an output variable (for current) in a circuit. It also calculates input and output resistances. Any nonlinear models are first linearized based on the DC operating point and then small-signal analysis is performed. The output variable can be any node voltage, while the input must be an independent source defined somewhere in the circuit.

Assumptions—Analog circuit, linear models. Models are linearized.

The DC small signal gain is the derivative of the output with respect to the input at the DC bias-point (and zero frequency). For example:

$$\frac{dV_{OUT}}{dV_{IN}}$$

The input and output resistance of a circuit refers to the “dynamic” or small-signal resistance at the input or output. Mathematically, small-signal DC resistance is the derivative of the input voltage with respect to the input current at the DC bias-point (and zero frequency). The following is an expression for input resistance:

$$\frac{dV_{IN}}{dI_{IN}}$$

In Multisim, the results of the **Transfer Function Analysis** produces a chart showing the ratio of the output to the input signal, the input resistance at the input source node and the output resistance across the output voltage nodes.

Note This is a DC analysis and does not calculate either the time- or frequency-domain transfer function.

Setting Transfer Function Analysis Parameters

Before you perform the analysis, review your circuit and decide on an output node, a reference node and an input source.

Set the **Transfer Function Analysis** parameters as desired:

- **Input source** drop-down list—Choose a voltage or current source.
- **Output node** drop-down list—For **Voltage** only, this is the point in the circuit for which you want to see results.
- **Output reference** drop-down list—For **Voltage** only, this is the reference voltage.

Transfer Function Analysis produces a chart showing the transfer function (output/input), input resistance at the input source and output resistance across the output voltage nodes or at the output variable.

Setting Transfer Function Analysis Parameters

Complete the following steps to set up and run **Transfer Function Analysis**:

1. Choose an input source from the **Input source** drop-down list
2. Enable **Voltage** and select an output node from the **Output node** drop-down list and an output reference node (usually ground or node 0) from the **Output reference** drop-down list

   Or

   Enable **Current** and select a current from the **Output source** drop-down list.

Note A source current or output node voltage can be used for this analysis.

You can filter the variables displayed to include internal nodes (such as nodes inside a BJT model or inside a SPICE subcircuit), open pins, as well as output variables from any submodules contained in the circuit. Filtering the variables shortens the list of results.

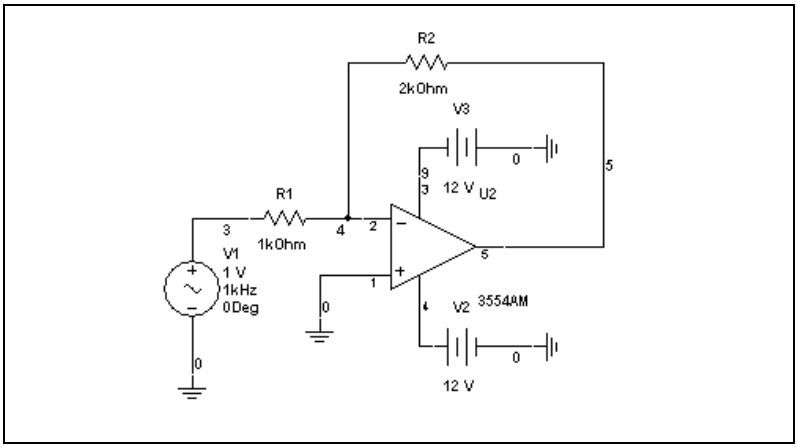
Complete the following steps to filter the variables displayed:

1. Click **Change Filter**. The **Filter Nodes** dialog box appears.
2. Enable one or more settings.
3. Click **OK**.

Refer to the *Working with Analyses* section for information about the other tabs in the analysis window.

Examples of Transfer Function Analysis

Linear Example



The circuit above is an inverting amplifier. The gain of this circuit is 2. We can define the mathematical function for this circuit as:

Vout = - 2Vin

Taking the derivative of the equation above, we get:

$$\text{Transfer function} = \frac{\partial V_{out}}{\partial V_{in}} = -2$$

Finding Rin

$$\because I_{in} \approx \frac{V_{in}}{R1}$$

Rearranging the equation we get:

$$R_{in} = R1 = \frac{V_{in}}{I_{in}}$$

Finding Rout

Since the Zin is much smaller than the op-amp impedance, Zout ~ 0.

Verifying the Results with Multisim

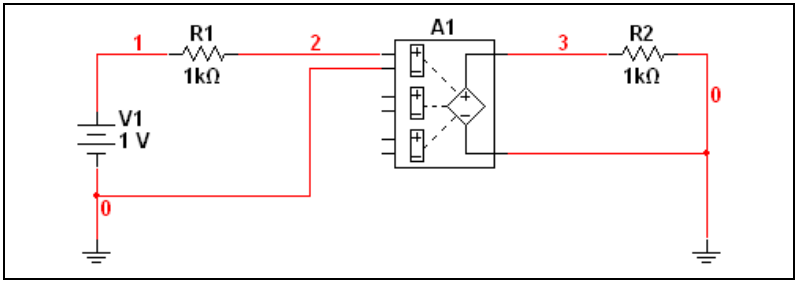
Complete the following steps to verify the results:

1. Set the following in the **Analysis Parameters** tab:
   - **Input source**—vv1
   - **Voltage** radio button—Enable
   - **Output node**—V(5)
   - **Output reference**—V(0)
2. Click **Simulate**. Note that the transfer function result shown below closely matches the calculated value.

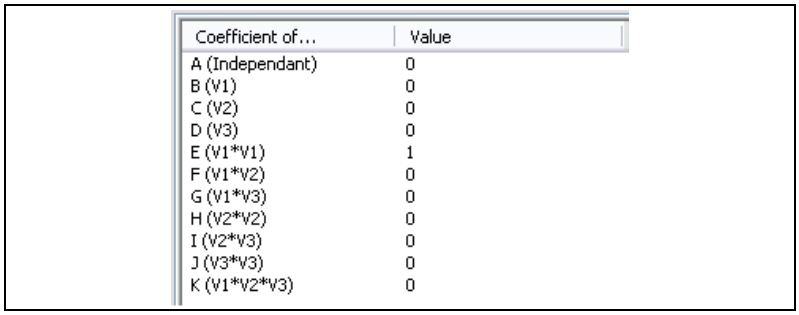
Transfer_Linear
Transfer Function

	Transfer Function Analysis	
1	Transfer function	-1.99994
2	vv1#Input impedance	1.00002 k
3	Output impedance at V(V(5),V(0))	599.98402 u

Non-linear Example



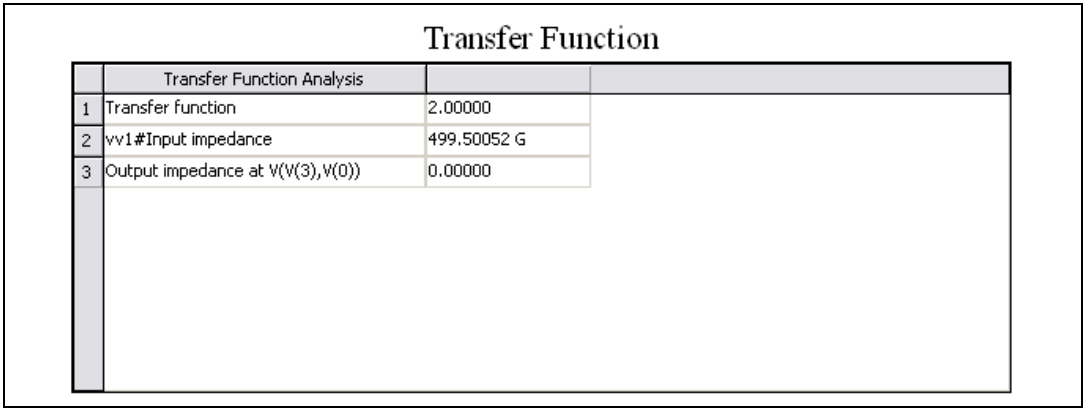
This example uses the polynomial source (A1) to square the input voltage. Double-click on A1 and enter “1” for the value of the coefficient of E as shown below. All other fields should be 0.



Coefficient of...	Value
A (Independant)	0
B (V1)	0
C (V2)	0
D (V3)	0
E (V1*V1)	1
F (V1*V2)	0
G (V1*V3)	0
H (V2*V2)	0
I (V2*V3)	0
J (V3*V3)	0
K (V1*V2*V3)	0

Complete the following steps to run the analysis:

1. Set up the analysis as follows:
   - **Input source**—vv1
   - **Output node**—V(3)
   - **Output reference**—V(0)
2. Click **Simulate** to run the simulation.



Transfer Function

	Transfer Function Analysis	
1	Transfer function	2.00000
2	vv1#Input impedance	499.50052 G
3	Output impedance at V(V(3),V(0))	0.00000

Verifying the Results

The equation for the circuit is:

$$V_{out} = V_{in}^{2}$$

Therefore:

$$\text{Transfer function} = \frac{\partial V_{out}}{\partial V_{in}}$$
$$= 2V_{in}$$
$$= 2(1)$$
$$= 2$$

Note that this value matches the value in the chart that was produced during the simulation.

Worst Case Analysis

Worst Case Analysis is an analysis that lets you explore the worst possible effects of variations in component parameters on the performance of a circuit. Multisim performs the **Worst Case Analysis** in conjunction with a DC or AC analysis. In either case the simulation is first performed with nominal values. Next, a Sensitivity analysis (either AC or DC) is performed to determine the sensitivity of the specified components on the output voltage or current. Lastly, the simulation is performed with the component parameter values that will produce worst case values at the output. The worst case parameters are determined by adding or subtracting the tolerance value from the nominal value according to whether the component's sensitivity on the output is a positive or negative value.

Assumptions—Applied to an analog circuit, DC and small-signal. Models are linearized.

Worst Case Analysis – DC Analysis

Upon selecting DC analysis the following calculations are performed:

1. DC sensitivity. If the DC sensitivity of the output voltage with respect to a specific component is determined to be a negative number, then the minimum value of this component is calculated. For example, if the DC sensitivity of resistor R1 is –1.23 V/Ohm, then the minimum value is derived from the following formula:

   $R1_{min} = (1 - \text{Tolerance}) \times R1_{nom}$,

   where,

   $R1_{min}$ = the minimum value of the resistor R1

   Tolerance = Tolerance specified by the user. (Tolerance is either an absolute value, or a percentage of the nominal value.)

   $R1_{nom}$ = Nominal value of the resistor R1

If the DC sensitivity of the output voltage with respect to a specific component is determined to be a positive number, then the maximum value of this component is calculated. For example, a resistor with a positive sensitivity would be determined according to the formula:

$R2_{max} = (1 - \text{Tolerance}) \times R2_{nom}$,

where,

$R2_{max}$ = the maximum value of the resistor R2

Tolerance = Tolerance specified by the user, expressed as a fraction of 1

$R2_{nom}$ = Nominal value of the resistor R2

2. The DC analysis is performed using the nominal value and either the minimum or maximum value of the resistance according to the sign of sensitivity analysis.

Worst Case Analysis – AC Analysis

Upon selecting AC analysis the following calculations are performed:

1. AC sensitivity to determine the sensitivity of the components on the output voltage.
2. According to the sensitivity results the minimum or maximum values of the selected components are calculated as explained above.
3. AC analysis is performed using the above calculated values for the components.

Setting Tolerance Parameters

Before you perform the analysis, review your circuit and decide on an output node.

In the **Model tolerance list** tab, choose which tolerance parameters are to be used. You can do this using any of the following methods:

To edit a tolerance in the list, select it and click **Edit selected tolerance**. The tolerance's current variable settings appear. Modify the variables as desired and click **OK** to save.

To delete a tolerance from the list, select it and click **Delete selected tolerance**.

1. To add a tolerance, click **Add tolerance**. The **Tolerance** dialog box appears.

2. Enter the following parameters as required:
   - **Parameter Type**—Choose type of sweep to be performed: **Model Parameter** or **Device Parameter**.
   - **Device Type**—Choose desired device from list.
   - **Tolerance Type**—Choose **Absolute** to enter a value, or **Percent** to vary the parameter by the specified percentage of its stated value.
   - **Toldrance value**—Enter a percentage value, or a set value depending on the tolerance type selected.
3. Enter the desired variables in the appropriate fields.

Setting Worst Case Analysis Parameters

Set the **Worst Case Analysis** parameters as follows:

- **Analysis** drop-down list—Choose **DC Operating Point** or **AC Analysis**. For details on either parameter, see *DC Operating Point Analysis*.
- **Output variable** drop-down list—Choose an output variable.
- **Collating Function** drop-down list—Choose a function: MAX, MIN, RISE_EDGE, FALL_EDGE, FREQUENCY.
- **Direction** drop-down list—Choose a direction: Low or High.
- **Change Filter** button—Click to change the filter that affects the list of possible output variables.
- **Expression** checkbox—Enable to enter an analysis expression in the **Output Variable** field.
- **Group all traces on one plot** checkbox—Enable as desired.

For DC circuits, the worst case analysis generates a table of the circuit's possible output voltages ranging from the nominal specification value to the worst case value. A list of the components and their worst case values appears in tabular form.

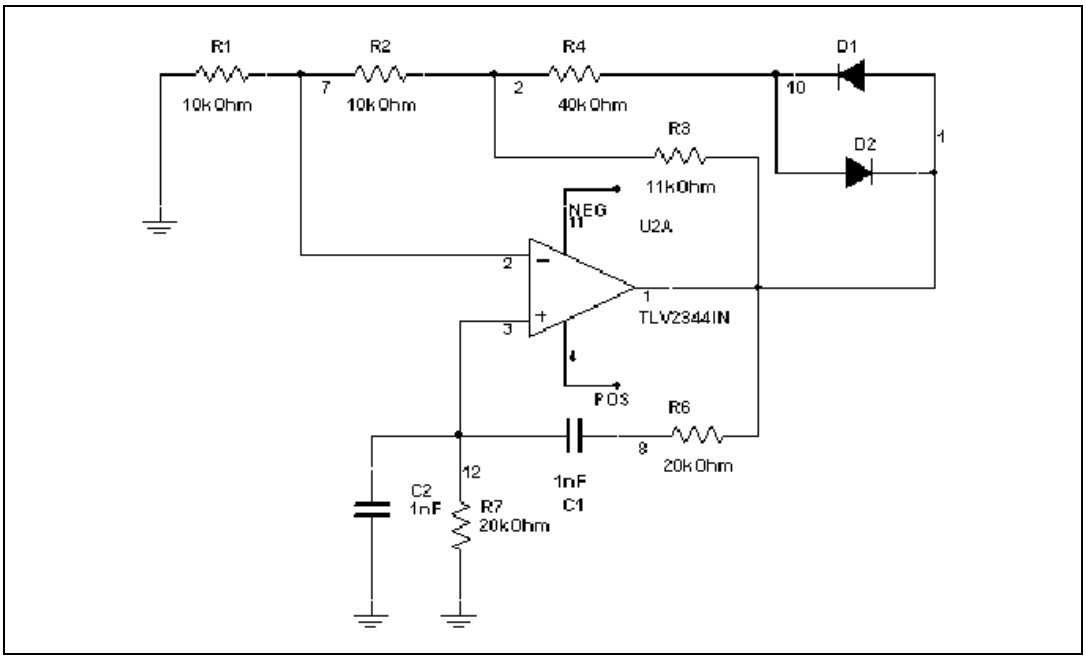
For AC circuits, the worst case analysis generates separate plots for the nominal and worst case runs. A list of the components and their worst case values appears in tabular form.

Analysis Expressions—As well as being able to select a node in the **Output Variable** field, you can click on the **Expression** checkbox and enter an analysis expression in the **Output Variable** field. When the **Expression** checkbox is enabled, the **Change Filter** button changes to the **Edit Expression** button. You can click this to view the **Analysis**

Expression dialog box where you can enter and edit expressions. Refer to the *Adding Analysis Expressions* section for more information.

Worst Case Analysis Example

This circuit is a Wien-bridge oscillator. The feedback to the non-inverting pin is designed to stabilize the amplitude of the oscillator. R1 – R4 must be chosen carefully to ensure this circuit will oscillate.



This circuit will oscillate if the following conditions are met:

$$\frac{R_2 + R_3}{R_1} > 2 \quad \text{and} \quad \frac{R_2 + R_3 \parallel R_4}{R_1} < 2$$

Under normal conditions, the output is a sinewave of approximately 8 v p-p.

Complete the following steps to set up the analysis:

1. Open the **Worst Case Analysis** dialog box and click **Add tolerance**.
2. Enter the parameters displayed below in the **Model tolerance list** tab.

Model	Parameter	Tolerance
r:xr3	resistance	Instance value:5%
r:xr1	resistance	Instance value:5%
r:xr2	resistance	Instance value:5%
r:xr4	resistance	Instance value:5%

3. Click on the **Analysis Parameters** tab and confirm the following settings:
   - **Analysis**—DC Operating Point
   - **Output variable**—V(1)
   - **Direction**—High
   - **Group all traces on one plot**—Enabled.
4. Click **Simulate**. The following results display.

Worst Case Analysis

$1, Nominal Run	-19.18902µ
$1, Worst Case Run	-17.94809µ

Run Log Descriptions

```
Descriptions of the runs

Tolerance changes needed to achieve worst case:
    r:xr3 resistance decreased to 10450
    r:xr1 resistance increased to 10500
    r:xr2 resistance decreased to 9500
    r:xr4 resistance decreased to 14250
```

The **Worst Case Analysis** section shows the DC operating point at node 1 for both nominal and worst case.

The **Run Log Descriptions** section displays the resistance values required to achieve the greatest difference from the nominal value.

Recall that one condition for oscillation is:

$$\frac{R_2 + R_3}{R_1} > 2$$

Inserting the output results, we get:

$$\frac{9.5 + 10.45}{10.5} = 1.9$$

This does not meet the oscillation requirement. Therefore the circuit may fail if the component tolerances are not carefully chosen.

Pole Zero Analysis

Pole Zero Analysis is used to determine the stability of electronic circuits by calculating the poles and zeros of the circuit's transfer function.

Note A pole is any number that will cause the denominator in a transfer function formula to be zero. A zero is any number that will cause the numerator in a transfer function formula to be zero.

A transfer function formula is a convenient way of expressing the behavior of analog circuits in the frequency domain. A transfer function is a Laplace Transform ratio of the output signal vs. input signal. The Laplace Transform of the output signal is commonly referred to as $V_O(s)$ and the Laplace Transform of the input signal is referred to as $V_I(s)$. The parameter s stands for $s=j\omega$, more commonly known as $s=j2\pi f$.

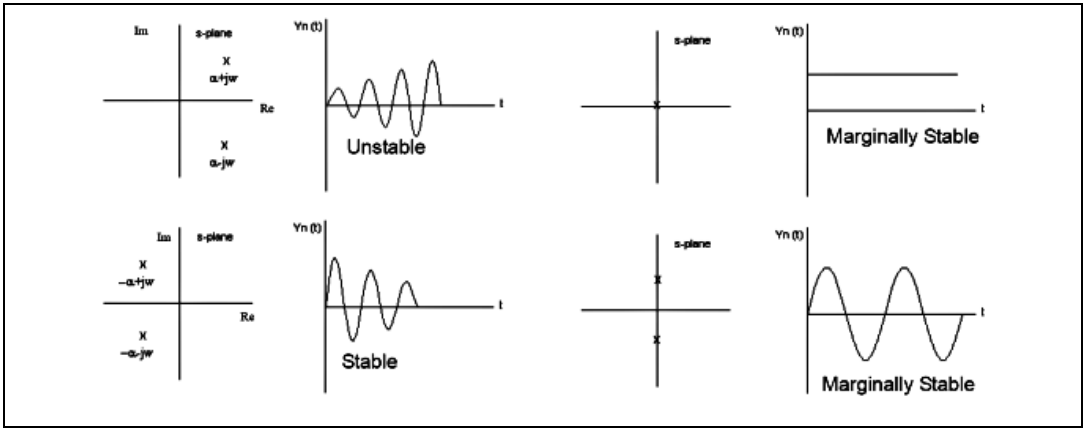
A transfer function is generally a complex value given by the magnitude response (or transmission) and phase response. The Transfer Function of a circuit can be expressed by the following formula:

$$T(s) = \frac{V_o(s)}{V_I(s)} = \frac{K(s+z_1)(s+z_2)(s+z_3)(s+z_4)\ldots}{(s+p_1)(s+p_2)(s+p_3)(s+p_4)\ldots}$$

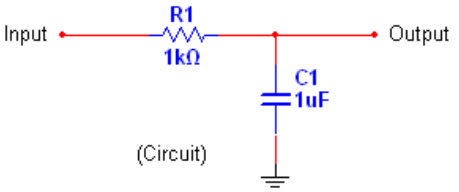
The numerator of the above formula contains the zeroes of the function ($-z_1,-z_2,-z_3,-z_4,...$) while the denominator of the function contains the Poles of the function ($-p_1,-p_2,-p_3,-p_4,...$).

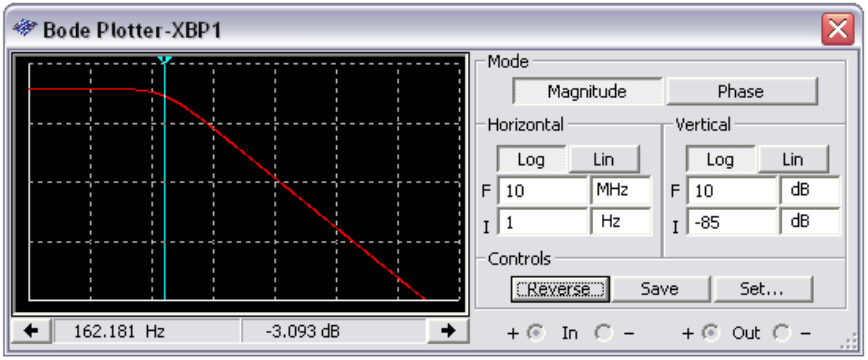
The zeroes of the function are those frequencies at which the transmission will be zero while the poles of the function are the natural modes of the network which define natural frequencies. Both, poles and zeroes can contain either real, complex or purely imaginary numbers.

Determining the poles and zeroes from a transfer function formula will allow the designer to predict how a circuit design will perform under operation. Since pole and zero values are numeric (real or imaginary), it is important to understand how these numbers relate to circuit stability. Refer to the figure below to see how different poles affect circuit stability in response to a step impulse. (“X” is the pole or zero value).



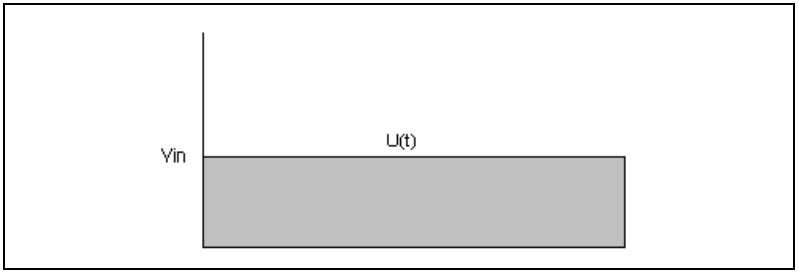
To illustrate this, analyze the following circuit.





(Circuit frequency response)

Assuming this passive filter performs as expected in the frequency domain, we will examine how this circuit will react when a signal is injected to the input. Since we are only interested in the initial response of the circuit, it is logical to test the circuit's response using a step function.



Analyzing the circuit, the following can be determined:

$$V_{out} = \left(\frac{Z_C}{R_1 + Z_C}\right) V_{in} \quad \text{where } Zc \text{ is the capacitive reactance}$$

and

$$Z_C = \frac{1}{C1_s}$$

The transfer function Vout/Vin:

$$\frac{V_{out}}{V_{in}} = \frac{\frac{1}{C1_s}}{R_1 + \frac{1}{C1_s}} = \frac{1}{R_1 C1_s + 1}$$

If C = 1uf and R = 1Kohm, the following can then be determined:

$$\frac{V_{out}}{V_{in}} = \frac{1}{0.001s + 1} = \frac{1000}{s + 1000}$$

A pole is any number that will cause the denominator in a transfer function formula to be zero. In this case:

$$0 = s + 1000$$
$$s = -1000$$

Zeroes are any number that will cause the numerator in a transfer function formula to be zero. There are no zeroes in the above transfer function formula since the numerator does not contain an “s” term.

Referring back to the first figure in this section, it is clear that this circuit is not unstable since the only pole in the above transfer function exists in the negative region of the s-plane.

Multisim Approach

Pole Zero Analysis computes the poles and/or zeroes in the small-signal AC transfer function. The program first computes the DC operating point and then determines the linearized, small-signal models for all the nonlinear devices in the circuit. This circuit is then used to find the poles and zeroes of the transfer function.

- output voltage / input voltage
- output voltage / input current

The result is a real and/or imaginary coordinate of the poles and/or zeroes, depending on which analyses are enabled.

Note The **Pole Zero Analysis** provides precise results on circuits containing passive devices (resistors, capacitors and inductors). Circuits containing active devices (transistor or op-amps) will not always display the expected results.

Setting Pole Zero Analysis Parameters

Before you perform the analysis, review your circuit and decide on input and output nodes (positive and negative). The input nodes are the positive and negative points in the circuit which are the transfer function inputs. Likewise, the output nodes are the positive and negative points in the circuit which are the transfer function outputs. You can use 0 (ground) for both positive nodes or both negative nodes.

Set the **Pole Zero Analysis** parameters as follows:

- **Analysis Type** box—Choose one of **Gain Analysis**, **Impedance Analysis**, **Input Impedance**, or **Output Impedance**.
- **Nodes** box—Choose the desired positive and negative **Input** and **Output** nodes.
- **Analyses performed** drop-down list—Choose analyses to be performed: **Pole Analysis** (finds poles of transfer function), **Zero Analysis** (finds zeroes of transfer function), **Pole and Zero Analysis** (finds both).

Pole Zero Analysis produces the real and imaginary coordinates of the poles and/or zeroes, depending on which analyses are enabled.

The **Pole Zero Analysis** provides precise results on circuits containing passive devices (resistors, capacitors and inductors). Circuits containing active devices (transistor or op-amps) will not always display the expected results.

Setting Pole Zero Analysis Parameters

For normal use, you only need to:

- select the analysis type by enabling the desired type
- select an input node from the **Input (+)** drop-down list and the **Input (–)** drop-down list
- select an output node from the **Output (+)** drop-down list and the **Output (–)** drop-down list

- select the analysis to be performed, by choosing from the **Analysis performed** list

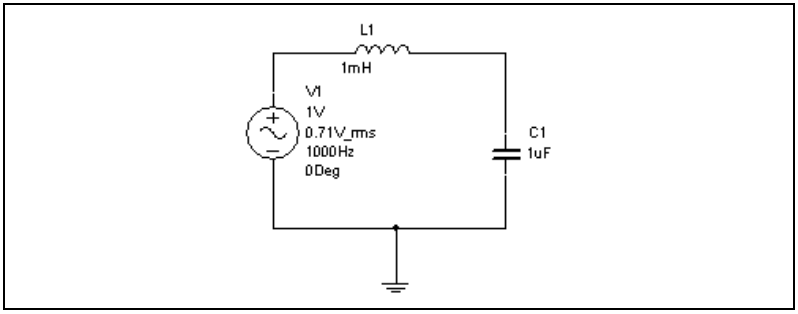
For more advanced use, you can filter the variables displayed to include internal nodes (such as nodes inside a BJT model or inside a SPICE subcircuits), open pins, as well as output variables from any submodules contained in the circuit.

Complete the following steps to filter the variables displayed:

1. Click **Change Filter**. The **Filter Nodes** dialog box appears.
2. Enable one or more settings.
3. Click **OK**.

Running Pole Zero Analysis

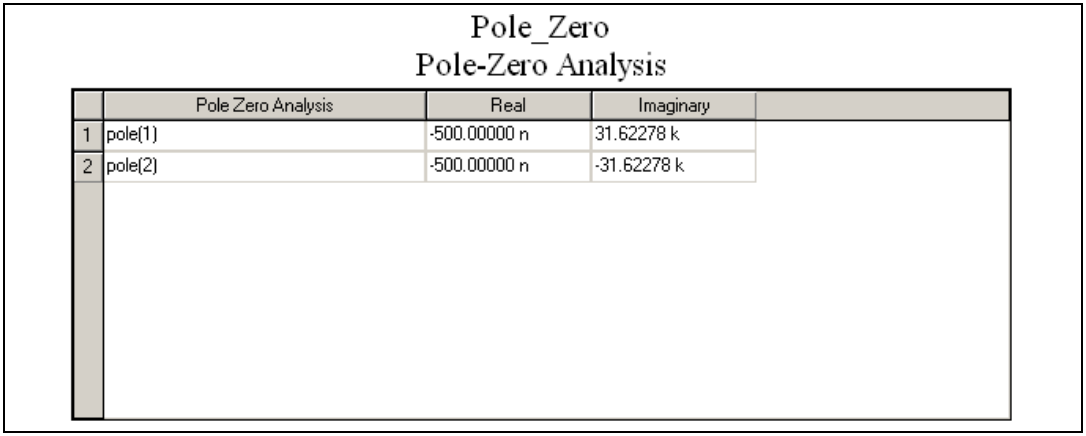
For this pole zero example, we will use the following circuit:



Complete the following steps to setup and run pole zero analysis using the circuit shown above:

1. Select **Simulate»Analyses»Pole Zero**.
2. In the **Pole-Zero Analysis** dialog, in the **Analysis Parameters** tab:
   - Set **Input (+)** to V(2) (input node).
   - Set **Input (–)** to V(0) (ground node).
   - Set **Output (+)** to V(1) (output node).
   - Set **Output (–)** to V(0) (ground node).
   - Set the **Analyses Performed** option to Pole And Zero Analysis.

3. Click **Simulate**. The **Grapher** appears displaying the following:



This graph indicates that two poles exist. One of the poles appears in the negative region and one in the positive region of the s-plane. Because one of the poles exists on the positive region of the s-plane, the stability of this circuit is poor.

Monte Carlo Analysis

Monte Carlo Analysis is a statistical technique that lets you explore how changing component properties affects circuit performance. **Monte Carlo Analysis** will perform **DC**, **AC** or **Transient Analysis** and vary the component properties. Multiple simulations are performed (also known as runs) and, for each run, the component parameters are randomly varied according to the distribution type and parameter tolerances specified by the user.

The first simulation is always performed with nominal values. For the rest of the simulations, a delta value is randomly added to or subtracted from the nominal value. This delta value can be any number within the *standard deviation* (σ). The probability of adding a particular delta value depends on the probability distribution. The two probability distributions available are:

- Uniform distribution (also know as Flat Distribution)
- Guassian distribution (also known as Normal Distribution)

Uniform Distribution

A uniform distribution is one for which the probability of occurrence is the same for all values of x. This could be the component value within a specified tolerance. For example, if a fair die is thrown, the probability of obtaining any one of the six possible outcomes is 1/6. Since all outcomes are equally probable, the distribution is uniform.

The distribution of a population can be described by several parameters such as the *mean* and *standard deviation.* The arithmetic mean is commonly called the average. The mean is the sum of all the observed properties divided by the number of observations.

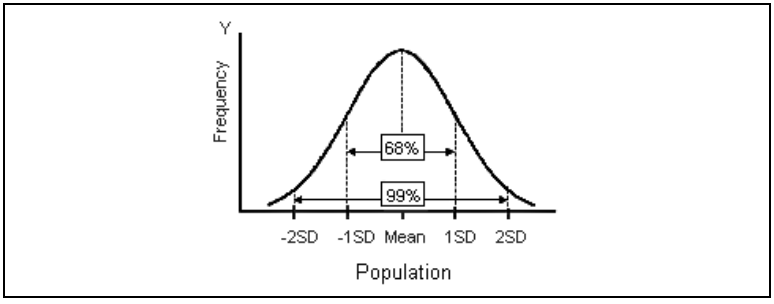
The formula in summation notation is:

$$\mu = \Sigma\frac{X}{N}$$

where μ is the population mean and N is the number of scores.

Gaussian Distribution

Many statistical tests assume a gaussian distribution. Most of these tests work well even if the distribution is only approximately normal and in many cases as long as it does not deviate greatly from normality. Gaussian distributions are a family of distributions that have the shape shown in the figure below.



Gaussian distributions are symmetric with most observed properties concentrated in the middle than in the tails. They are defined by two parameters: the mean (μ) and the standard deviation (also know as σ, SD or Sigma).

The formula for the height of a normal curve for a given value of x is:

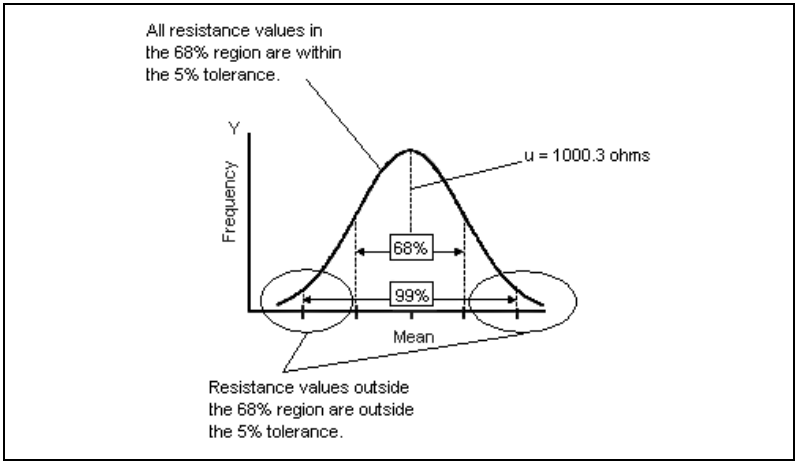
$$\frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-(x-\mu)^2/2\sigma^2}$$

The standard deviation (σ) is calculated from the parameter tolerance according to:

$$\sigma^2 = \frac{\Sigma(X-\mu)^2}{N}$$

In Multisim, gaussian distribution will insure that only 68% of the population is within the specified tolerance. The remaining population will be outside the tolerance (specified by the user). As an example, let us look at the gaussian distribution for a 1k resistor with 5% tolerance.

The standard deviation leads to a tolerance band of 50 Ω. Therefore, one standard deviation leads to a tolerance from 0.95kohms to 1.05 kΩ (1 kΩ +/– 50 Ω). Only 68% of the population will be within the tolerance of 5%. With a large enough sample, the μ (mean) will be approximately 1000 ohms.


All resistance values in the 68% region are within the 5% tolerance.
Y
Frequency
u = 1000.3 ohms
68%
99%
Mean
Resistance values outside the 68% region are outside the 5% tolerance.

Multisim approach

Multisim begins by performing a simulation using the nominal component values. For the rest of the simulations, a delta value is randomly added to or subtracted from the nominal value for each component/tolerance. This delta value can be any number within the standard deviation. The probability of adding a particular delta value depends on the probability distribution, which is selected by the user (for each component tolerance).

Setting Up and Running Monte Carlo Analysis

To start Monte Carlo Analysis, select **Simulate»Analyses»Monte Carlo**. The **Monte Carlo Analysis** dialog box displays.

Entering a Component Tolerance

Complete the following steps to enter component tolerances:

1. Click **Add tolerance** in the **Model tolerance list** tab. The **Tolerance** dialog box displays.
2. Specify the following:
   - **Parameter Type**—Select Device Parameter in the drop-down list. This option allows you to select which component (R1, U5, etc) will be included in a tolerance. In addition, this section allows you to select the parameter to be used in this tolerance (eg: Resistance, Temperature Coefficient, etc).
   - **Distribution**—Select either Uniform or Gaussian.
   - **Tolerance Type**—This option allows the user to specify if the tolerance value is a percentage of the component value) of a specific amount of the component value (eg: Using a resistor, if the Tolerance Type is Absolute and the Tolerance Value is 20, the component tolerance will be +/–20 Ω).
   - **Tolerance value**—This option allows the user to set the magnitude of the tolerance (in percent or absolute depending on what is selected in the Tolerance Type option).
3. Click **Accept**.

Specifying Monte Carlo Analysis Parameters

Set the analysis parameters in the **Analysis Parameters** tab as described below:

Note Refer to the *Output Tab*, *Analysis Options Tab*, and *Summary Tab* sections for details on the other tabs in the analysis window.

1. **Analysis**—This option allows the user to specify what type of simulation Monte Carlo will perform for each run. There are three options:
   - Transient Analysis (time domain)
   - DC Operating Point (DC operating point of every node in the circuit)
   - AC Analysis (frequency domain)
2. **Number of runs**—This option allows the user to specify how many simulations Monte Carlo will generate. Each simulation will use a slightly different component value.
3. **Analysis Expressions**—As well as being able to select a node in the **Output Variable** field, you can click on the **Expression** checkbox and enter an analysis expression in the **Output Variable** field. When the **Expression** checkbox is enabled, the **Change Filter** button changes to the **Edit Expression** button. You can click this to view the **Analysis Expression** dialog box where you can enter and edit expressions. Refer to the *Adding Analysis Expressions* section for more information.
4. **Collating Function** and **Threshold**—There are four options available:
   - **MAX**—This option will inform the user (when the Monte Carlo results have been generated) what the maximum voltage (peak) is for each run.
   - **MIN**—This option will inform the user (when the Monte Carlo results have been generated) what the minimum voltage (peak) is for each run.
   - **RISE_EDGE**—This option will inform the user (after Monte Carlo has generated the results) the time when the signal reached the *threshold* voltage on the first rising edge of the waveform. As an example:

     A 1 KHz Sine wave generator is connected to a voltage divider. If the threshold setting is 0 (volts), Monte Carlo will generate a

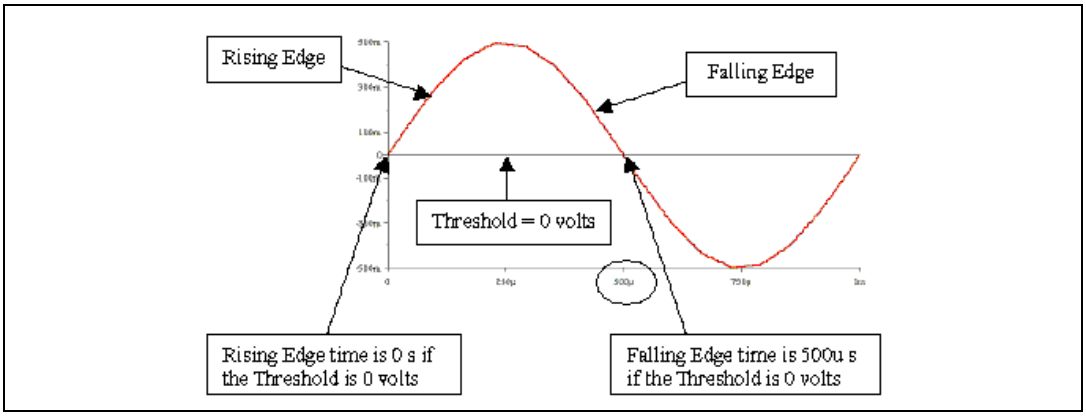
'worst case value' of zero (since the waveform exceeds the threshold voltage (0 volts) at zero seconds).

Refer to figure below.

- **FALL_EDGE**—This option will inform the user (after Monte Carlo has generated the results) the time when the signal reached the *threshold* voltage on the first falling edge of the waveform. As an example:

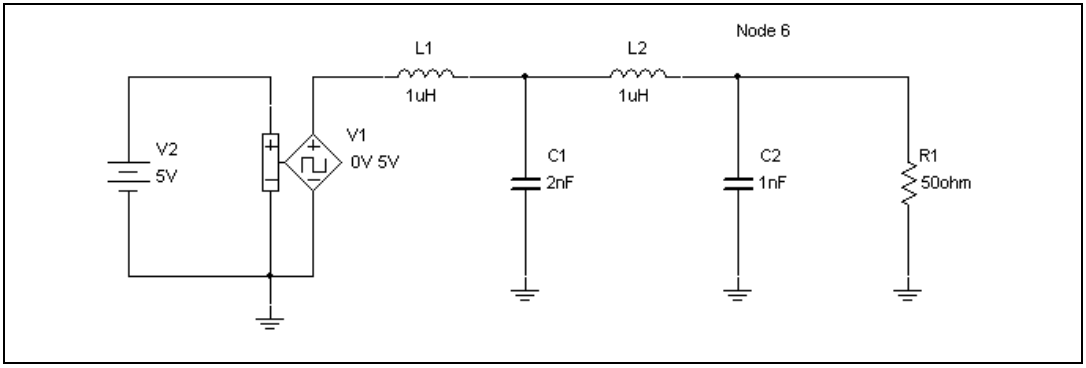
  A 1 KHz Sine wave generator is connected to a voltage divider. If the threshold setting is 0 (volts), Monte Carlo will generate a 'worst case value' of 0.001 since the waveform exceeds the threshold voltage on the negative edge at 0.0005 seconds (half of the period).

  Refer to the figure below.


Rising Edge
Falling Edge
Threshold = 0 volts
Rising Edge time is 0 s if the Threshold is 0 volts
Falling Edge time is 500u s if the Threshold is 0 volts

Monte Carlo Analysis Example

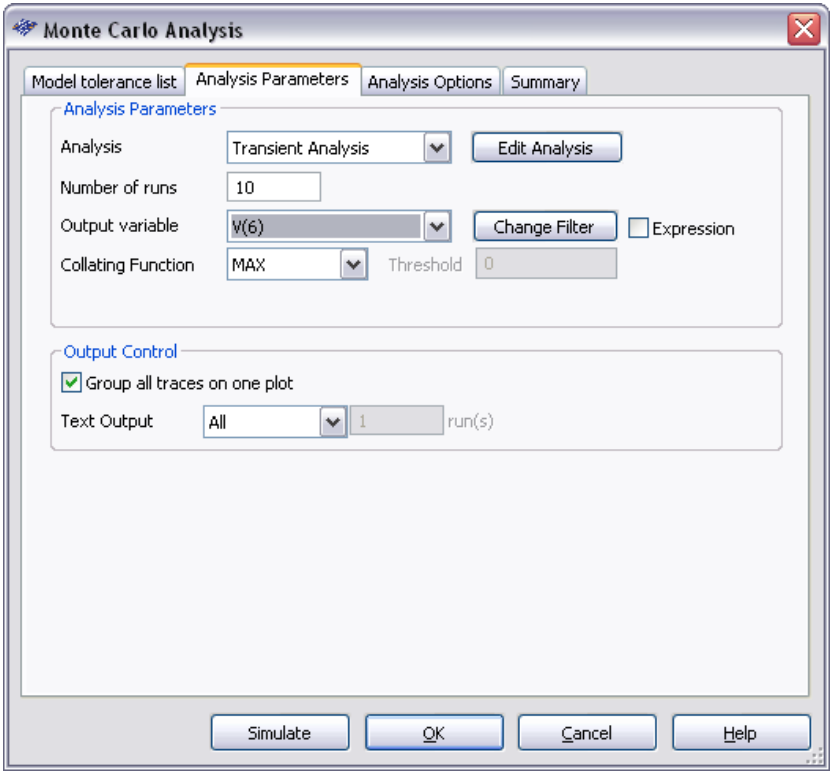
This example analyses the following circuit at node 6. Details follow.



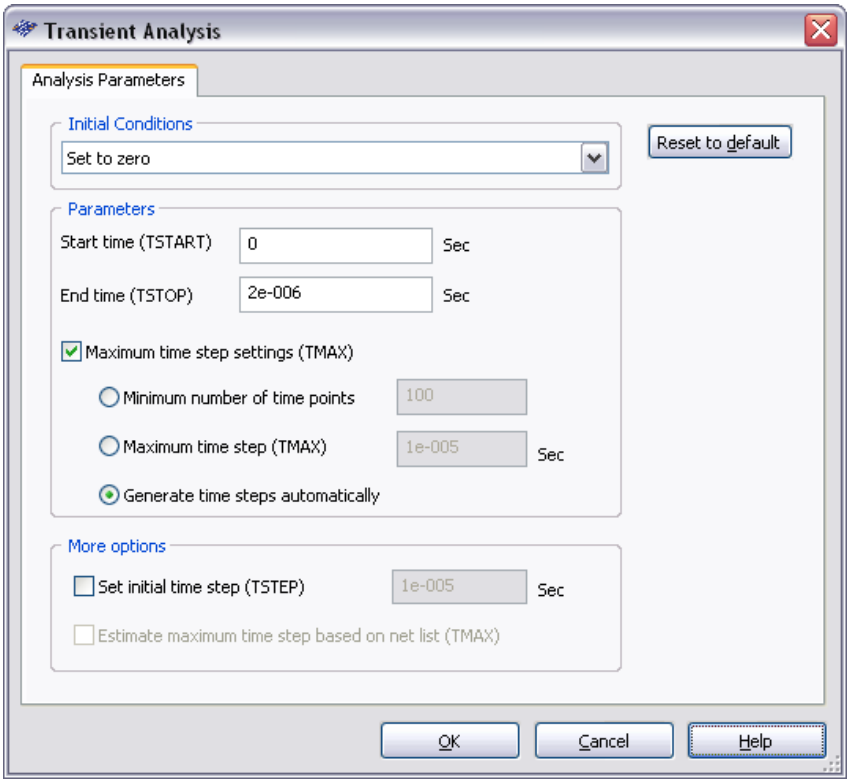
Setting up the Sample Monte Carlo Analysis

Complete the following steps to set up the sample analyses:

1. Fill in the data as shown below.

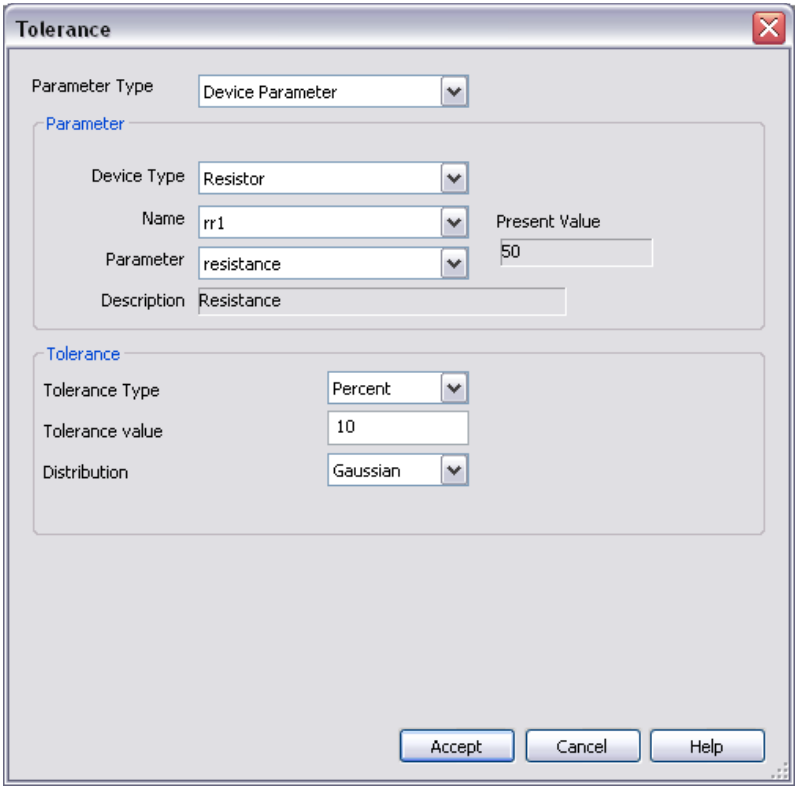


2. Click **Edit Analysis** to modify the transient analysis to be used for the simulation and change the parameters as shown below.

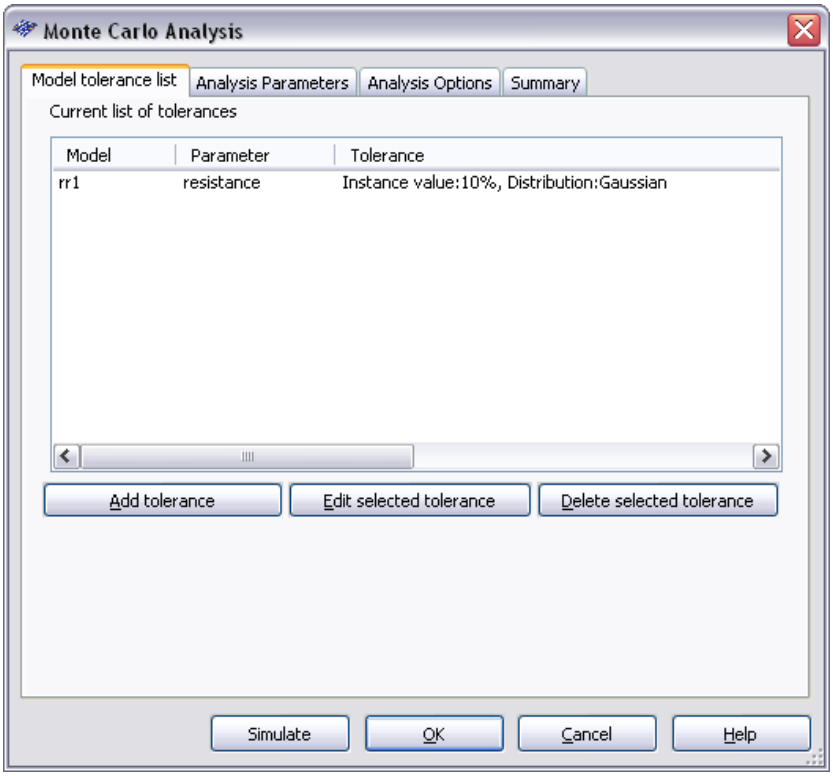


3. Click **OK**.
4. Click on the **Model tolerance list** tab, to specify component to vary during the simulation.

5. Click **Add tolerance** and enter all information in the fields as shown below.



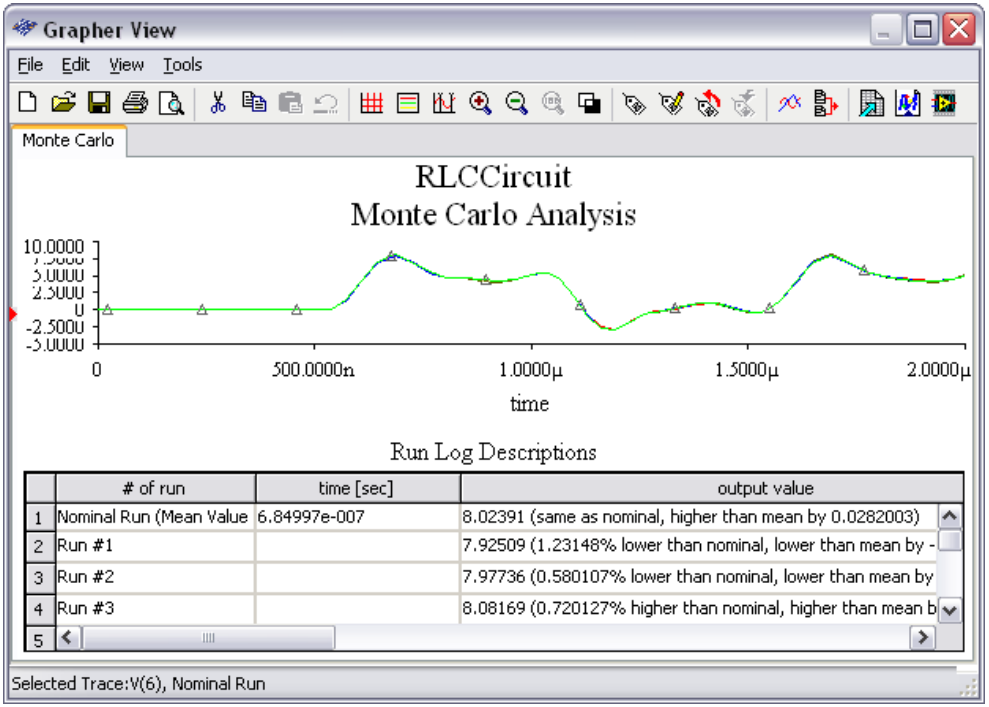
6. Click **Accept**. The model tolerance should appear as below.



Note To add more than one component to be varied during simulation, click **Add tolerance** again, and add new information.

7. Click **Simulate** to run the analysis.

Simulation Results



Note The results of your simulation may be different than what is shown above.

To export Voltage vs. Time data from the graph, click on the graph to make it active. Now click the **Excel** icon to export the data.

To export the data under **Run Log Descriptions** to Excel, place the mouse pointer within the data box and click once to make it active. Now click on the **Excel** icon. Excel should start automatically. See table below.

Run	Time (sec)	Voltage (V)	Sigma	Resistance (Ω)
Nominal	6.84997e-007	8.02391	0.0762047	50
1		7.99053	0.238791	49.3672
2		8.07372	0.166473	50.9656
3		7.99164	0.233375	49.3881

4		7.99839	0.200508	49.5152
5		7.66444	1.82732	43.8137
6		8.28861	1.21325	55.445
7		8.28717	1.20627	55.4133
8		7.68974	1.70408	44.2215
9		8.14702	0.523529	52.4342
10		8.27988	1.17076	55.2525

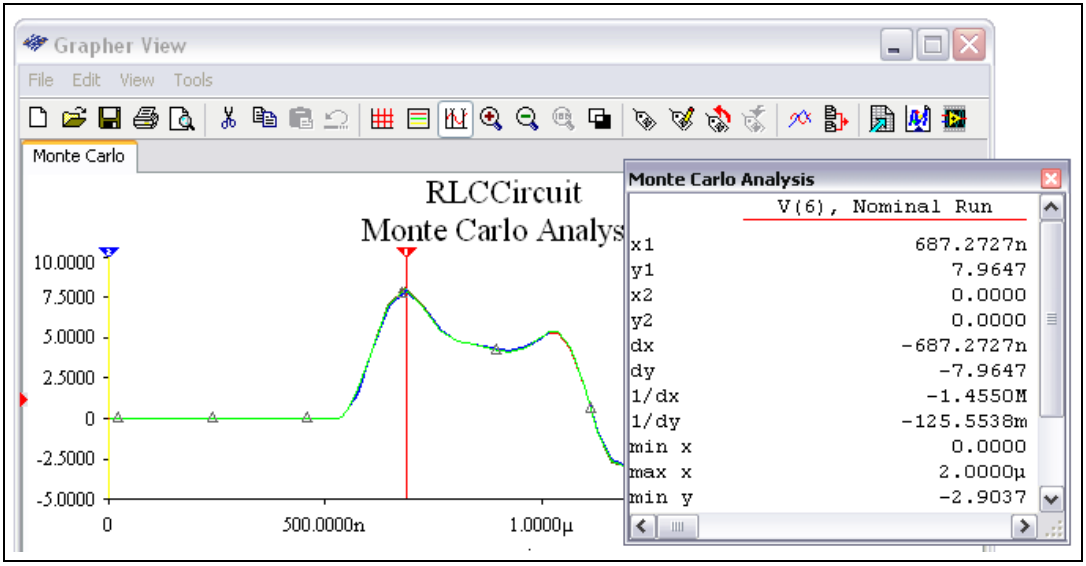
The data in the above table are extracted from **Run Log Descriptions**. The **Voltage** column refers to the first value under the **Worst value** column.

Output mean = 8.03955

Standard Deviation = 0.205279

The mean and standard deviation can be found under the **# of Run** column. You must expand this column to view this information.

The **Time** column in the above table indicates the instance where the voltage reaches maximum level. This is defined in the **Function** field in the **Analysis Parameters** tab.



The cursor in the above figure is approximately placed at the time where voltage reaches the max value (X1 in the cursor table). This is the same value that Multisim shows under voltage column (Y1 in the cursor table).

Calculating the Mean value from the table in the *Simulation Results* section:

$$\mu = \frac{\sum_{n-1}^{n} V_n}{n}$$

where V_n = voltage at each run

μ = mean value

n = number of runs (not including nominal run)

$$\mu = \frac{7.99 + \ldots + 8.27}{10} = 8.04 \text{ V}$$

Standard Deviation:

$$s^2 = \frac{\sum_{n\ 1}^{n} (V_n - \mu)^2}{n}$$

where s = standard deviation

V_n = voltage at each run

μ = mean value

n = number of runs (not including nominal run)

$$s = \sqrt{\frac{(7.99 - 8.04)^2 + \ldots + (8.27 - 8.04)^2}{10}} = 0.215$$

Finding **Sigma** for the table in the *Simulation Results* section:

Note sigma corresponds to resistance value data and not the voltage value from the above calculation. First we need to calculate the mean and standard deviation for the resistance data.

Finding the exact value for sigma requires the equation for the distribution curve. For our purpose we will use an approximation method to find sigma. We will use Run 10, since this will give us the closest value to the simulated results.

$$\mu = \frac{\sum_{n-1}^{n} R_n}{n}$$

where R_n = resistance at each run

μ = mean value

n = number of runs (not including nominal run)

$$\mu = \frac{49.36 + 50.96 + \ldots + 55.25}{10} = 50.58\ \Omega$$

At 1 standard deviation:

$$s^2 = \frac{\sum_{n\ 1}^{n} (R_n - \mu)^2}{n}$$

where R_n = resistance at each run

μ = mean value

n = number of runs (not including nominal run)

$$s = \sqrt{\frac{(49.36 - 50.58)^2 + (50.96 - 58.58)^2 + \ldots + (55.25 - 50.58)^2}{10}} = 4.03$$

Trace Width Analysis

Trace Width Analysis calculates the minimum trace width needed in the circuit to handle the RMS current at any trace/wire. The RMS current is derived from simulation. To fully understand the importance of this analysis, we must first understand what happens to a trace or wire as current increases on that wire.

The flow of current through a trace will cause the temperature of the trace to increase. The formula for power is $P= I^2R$, so the relationship is not simply linear with respect to current. The resistance of a trace (per unit length) is a function of its cross-sectional area (width times thickness). The relationship between temperature and current, therefore, is a non-linear function of current, trace width, and trace thickness. The ability of a trace to dissipate heat is a function of its surface area, or width (per unit length).

The PCB layout technology limits the thickness of the copper used for wires. This thickness is related to the nominal weight, which is provided in oz/ft^2, in the form of a table.

Understanding how trace width is determined

A common model in thermodynamics for current in a trace is:

$$I = K * \Delta T^{B1} A^{B2}$$

where:

I = current in amps

ΔT = change in Temperature above ambient, in degrees C

A = cross sectional area in square mils, and K, B1 and B2 are constants

This is the starting point for McHardy and Gandi.

To estimate the coefficients for the above equation, it is convenient first to convert them to linear form. We can do this by using logarithms, as follows:

$$\ln(I) = \ln(K) + B1 * \ln(\Delta T) +$$

where *ln( )* is the natural logarithm (to the base e).

DN Data

DN sources are charts relating temperature change and current for various trace configurations. The DN data provides information allowing the independent evaluation of length and width for the traces under study.

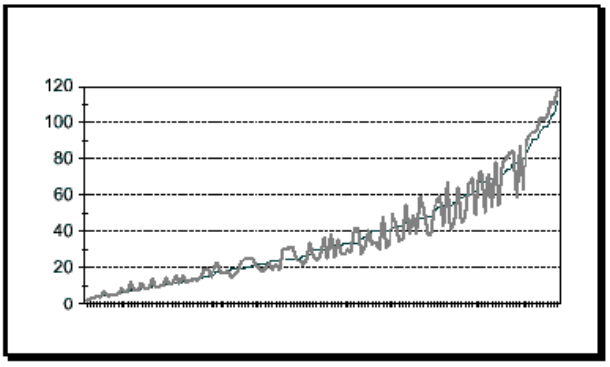
When all DN data is used in a regression analysis, we get the following estimate:

$$\ln(I) = -3.23 + 0.45 * \ln(\Delta T) + 0.69 * \ln(A)$$

which leads to:

$$I = 0.04 * \Delta T^{0.45} A^{0.69}$$

Below is a graph plotting approximately 300 data points from the above formula.



Graph of Actual vs Estimated Current from DN Data

Multisim Approach

Multisim uses the trace weight value (oz/ft^2) to calculate the required thickness for **Trace Width Analysis**. See the table below for the assumed trace thickness for each copper weight. Using transient analysis, the currents of each wire are calculated first. These currents are usually time dependent, that is, their amplitude changes in time to a positive or negative value.

Thickness	Width
1.0/8.0	.2
.0/4.0	.36
3.0/8.0	.52
1.0/2.0	.70
3.0/4.0	1
1	1.40
2	2.80
3	4.20
4	5.6
5	7.0

Thickness	Width
6	8.4
7	9.80
10	14
14	19.60

Since the transient analysis is performed for discrete time points, the accuracy of the maximum absolute value depends on how many time points are selected. Here are a few recommendations to increase the accuracy of **Trace Width Analysis**:

- Set the end time of the transient analysis (visible in the **Analysis Parameters** tab) to a time point where at least one cycle of the signal is processed. This is particularly the case if the signal is periodic. If not, you must set the end time to a value large enough for Multisim to capture the correct maximum current.
- Manually increase the number of points to 100 or more. The more points of the signal, the more accurate the maximum value. Note that increasing the number of time points beyond about 1000 will increase the execution time and slow down Multisim.
- Consider the effect of the initial condition, which can change the maximum of the signal at starting time. It may slow down the simulation if the steady state (DC operating point, for example) is far from the initial condition (say, zero IC).

Once I and ΔT are known, Multisim uses the McHardy and Gandhi formula to find the width of the wire. The formula is:

$$I = KT^{0.44}A^{0.725}$$

where:

I = maximum current in Amps

K = derating Constant (0.024 for inner)

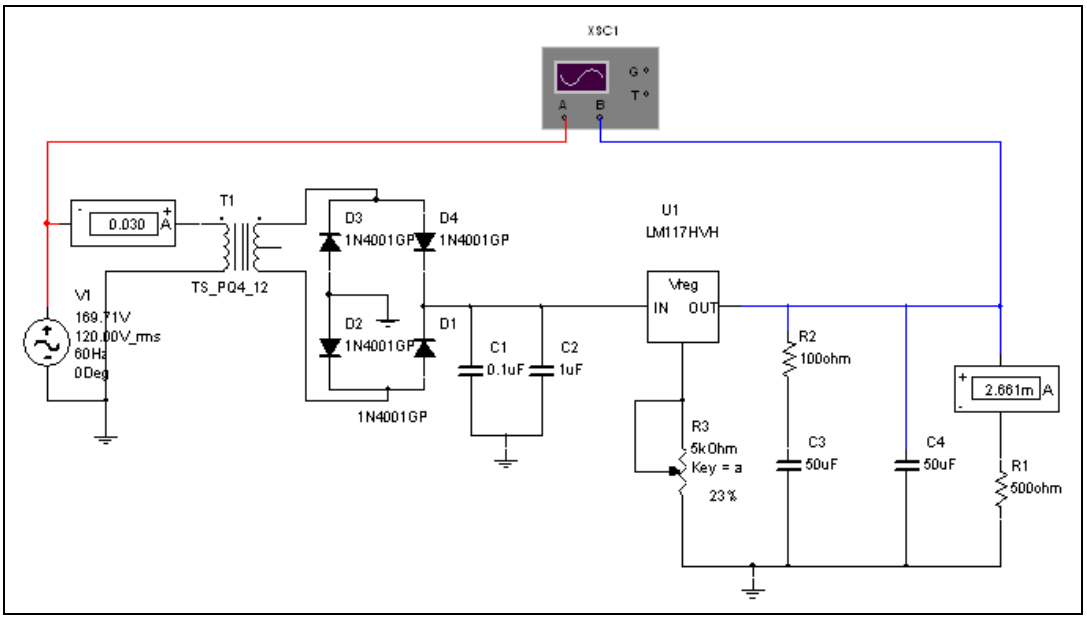
T = maximum temperature rise above ambient in × C

A = cross-sectional area in square mils (*not* millimeters)

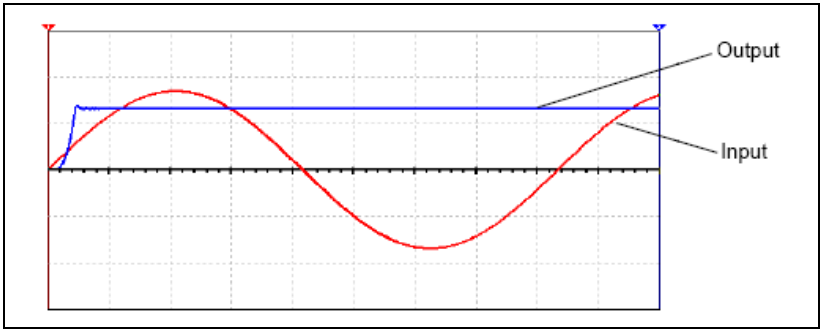
Note that one “mil” is 1/1000 of an inch.

Sample Trace Width Analysis

This example uses the following circuit:



Notice that the input signal is 120V AC (RMS). The input signal and the output signal are shown below using the Multisim **Oscilloscope**.



Complete the following steps to set up **Trace Width Analysis**:

1. Select **Simulate»Analyses»Trace Width Analysis**.
2. Select the **Trace Width Analysis** tab, and set the following:
   - **Maximum temperature above ambient** to 10 (degrees C).
   - **Weight of plating** to 1 (oz/ft^2).

- Optionally, enable **Set node trace widths using the results from this analysis**.
- **Units**—After a trace width analysis, the minimum trace width is output to the **Grapher**. Set the unit of measurement for the trace width here, typically mils.

3. Select the **Analysis Parameters** tab and choose **Set To Zero** from the **Initial Conditions** drop-down list.
4. Click **Simulate**. The **Grapher** displays with the results of the analysis in a chart. These results are listed below:

Component: c1,

- Pin #1, Trace Width = 0.270535 mils
- Pin #2, Trace Width = 0.270535 mils

Component: c2,

- Pin #1, Trace Width = 0.270535 mils
- Pin #2, Trace Width = 0.270535 mils

Component: c3,

- Pin #1, Trace Width = 0.0819493 mils
- Pin #2, Trace Width = 0.0819493 mils

Component: t1,

- Pin #1, Trace Width = 3.60528 mils
- Pin #3, Trace Width = 3.60528 mils
- Pin #4, Trace Width = 0.465827 mils
- Pin #5, Trace Width = 0.465827 mils

Trace Width Analysis determines that the minimum trace width required to pin 1 of T1 (transformer) is 4.07095 mils. Therefore, in order to insure that the connection between the input source and transformer T1 operates successfully, the trace width must be a minimum of 4.0 mils when the PCB board is created.

RF Analyses

RF analyses (Characterizer, Noise Figure and Matching Networks analyses) are performed through the Network Analyzer. Refer to the *RF Analyses* section of Chapter 14, *RF*, for more information.

Nested Sweep Analyses

Temperature Sweep and **Parameter Sweep** analyses can be performed in a nested fashion, with a series of sweeps being performed, each within the constraints of the sweep before it. For example, you can perform a temperature sweep on the results of a parameter sweep.

Complete the following steps to perform a **Nested Temperature** or **Parameter Sweep Analysis**:

1. Open the **Parameter Sweep** dialog box by choosing either **Temperature Sweep** or **Parameter Sweep** from the **Analyses** menu.
2. From the **Analysis to Sweep** drop-down list, select Nested Sweep.
3. Click **Edit Analysis**. The **Nested Parameter Sweep** dialog box appears. The title bar of the dialog box indicates you are defining nested sweep level 1.
4. Set the parameters as desired.
5. To create another level of the nest, again select Nested sweep from the **Analysis to sweep** drop-down list.
6. Click **Edit Analysis**. A new **Nested Parameter Sweep** dialog box appears, this time indicating that you are at nest Level 2.
7. You can continue to add nested sweeps by repeating this procedure.
8. To return to the higher level, saving your changes, click **OK**. To return to the higher level without saving your changes, click **Cancel**.
9. When all nested analyses have been defined, click **Simulate**.

Batched Analyses

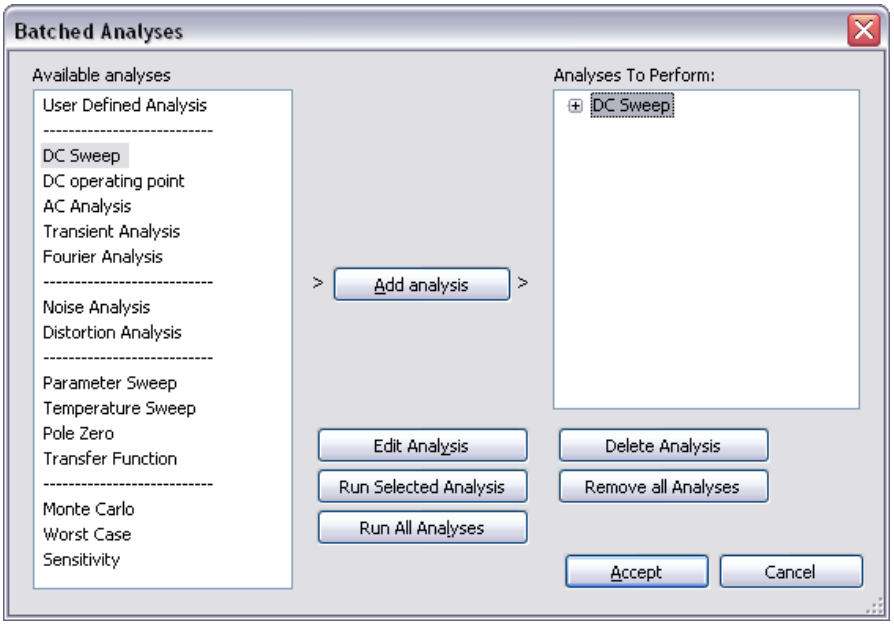
You can batch together different analyses, or different instances of the same analysis, to be performed in sequence. This provides a convenient way for advanced users to perform multiple analyses from a single, interpreted command.

For example, you might use batched analyses to:

- repeatedly perform the same set of analyses, such as when trying to fine-tune a circuit
- build a record of the analyses that you performed on the circuit
- set up a sequence of long running analyses to run automatically.

Complete the following steps to set up batched analyses:

1. Choose **Simulate»Analyses»Batched Analyses**. The **Batched Analyses** dialog box appears.



2. To add an analysis to the batch, select it and click the **Add Analysis** button. The parameters dialog box for the selected analysis appears, so you can set the parameters for the analysis.
3. When you have finished the settings for the analysis, click **Add to List**.
4. The analysis is added to the **Analyses to Perform** list on the right. Summary information can be revealed by clicking the "+" beside the analysis.
5. Continue to add analyses as desired. Note that the settings for one instance of an analysis become the default settings for that analysis during this operation. For example, if you set your first DC Sweep to an increment of 0.6, the 0.6 increment is the default value when you add your next DC Sweep to the batch.
6. To run just one of the analyses in the batch, select it and click **Run Selected Analysis**. To run all of them, click **Run All Analyses**.

To edit an analysis' parameters in the batch, select it and click **Edit Analysis**. The selected analysis' parameters dialog box appears, allowing you to make any modifications you wish to the analysis.

To remove an analysis from the batch, select it and click **Delete Analysis**.

To remove all analyses, click **Remove All Analyses**.

User Defined Analyses

The **User Defined Analysis** allows you to manually load a SPICE card or netlist and type in SPICE commands. This gives you more freedom to adjust a simulation rather than using the graphical interface of Multisim. However, a thorough knowledge of SPICE is essential.

To demonstrate the **User Defined Analysis**, two Berkeley SPICE 3F5 compatible SPICE netlists are simulated in the following sections.

Creating and Simulating a SPICE Netlist

This example will create and simulate the following SPICE netlist:

```
*Basic RC Circuit
v1 1 0 sin(0 1 1000)
r1 1 2 1000
c1 2 0 1e-6
.tran 0.1m 1m
```

Complete the following steps to create a SPICE netlist:

1. Type the netlist into any text editor, as in the example shown above.
2. From the text editor, select **File»Save As**.
1. In the dialog box that appears, enter `Rc.cir` for the filename and select `C:\Temp` for the filepath.

Complete the following steps to run the **User Defined Analysis**:

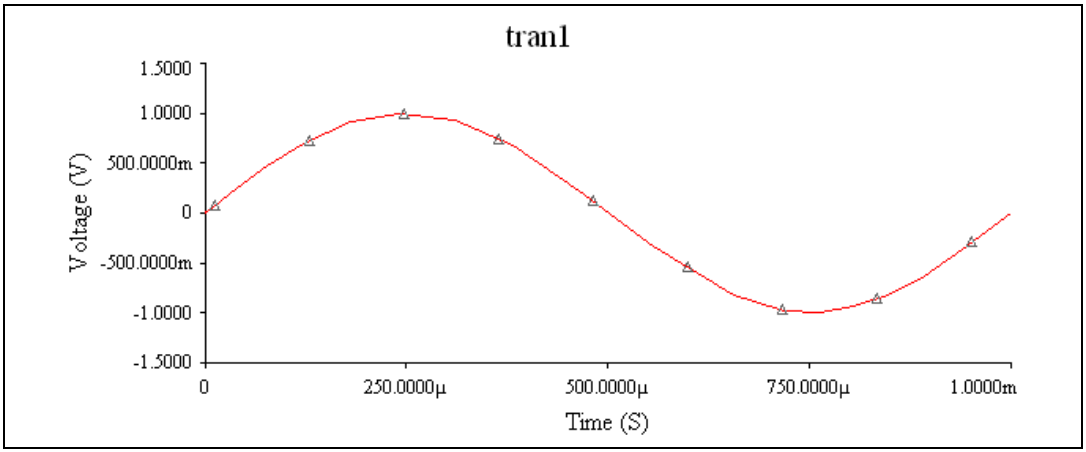
1. Select **Simulate»Analyses»User Defined Analysis**.
2. In the **Commands** tab of the **User Defined Analysis** dialog box, enter the following syntax:

   ```
   source C:\Temp\Rc.cir
   tran = 100u 1m
   plot v(1)
   ```

3. Click **Simulate**.

Note Multisim gives you the option of using an equal sign after the statement i.e. tran = 100u 1m or tran 100u 1m. Both revisions of the “tran” statement work. In addition, if you would like to have a smoother waveform, reduce the step value from 100u to 1u.

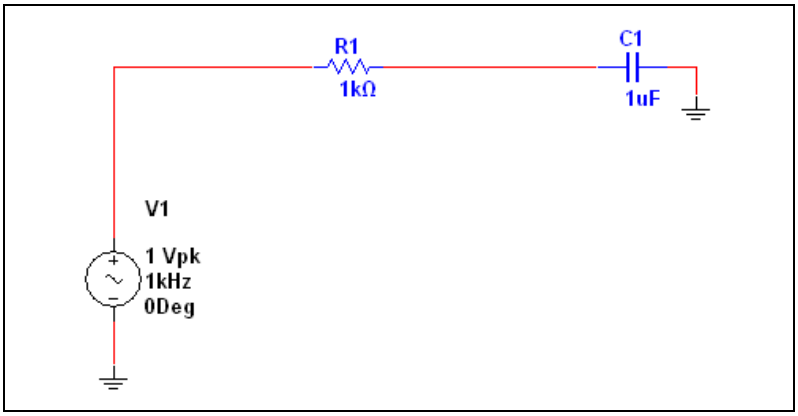
The waveform example in the figure below will appear on the your display grapher in Multisim.



Importing the SPICE netlist into Multisim

Complete the following steps to import the `Rc.cir` SPICE netlist into Multisim:

1. Select **File»Open**. Click on the drop-down arrow next to "Files of type", and select SPICE Netlist Files (`*.cir`).
2. Select the file `Rc.cir` from the `C:\Temp\` directory and click **Open**. The `Rc.cir` file imports into Multisim and shows the schematic equivalent of the text-based SPICE netlist.

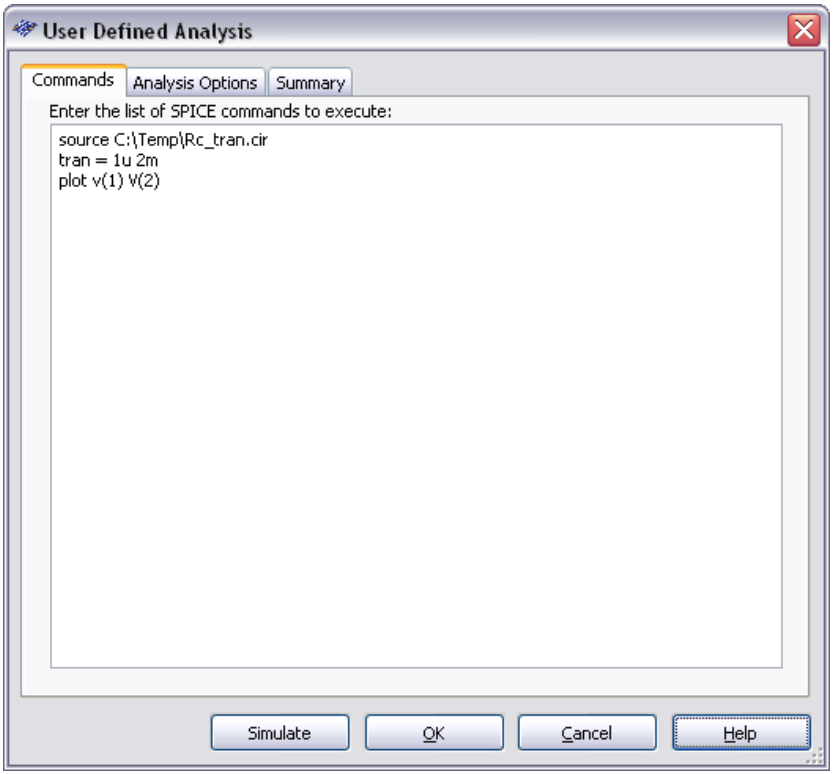


3. Select **File»Save As** and name the file `Rc.ms10`.

Plotting Two Nodes Using the Tran Statement

To plot two nodes using the tran statement enter the syntax shown below:

- First line shows the path and loads the SPICE netlist file
- Second line shows the "`tran`" statement with a 1u step value and a final stop value of 2m
- plot v(1) v(2) will plot nodes 1 and 2 on the **Grapher**.



Custom Analysis Options Dialog Box

Multisim lets you control many aspects of the simulation used within the analyses, such as resetting error tolerances, selecting simulation techniques and viewing the results. Simulation efficiency is also dependent on the options you choose.

This section briefly describes the simulation options you have for controlling simulation used within the analyses and lists their default

values. You will find these options through the **Analysis Options** tabs of the various analyses dialog boxes. Refer to the *Analysis Options Tab* section for more information.

Global Tab

Code	Option Name	Description	Default	Unit	Recommendation
ABSTOL	Absolute error tolerance	Resets the absolute current error tolerance. Default is suitable for most bipolar transistor VLSI circuits.	1.0e-12	A	—
VNTOL	Voltage error tolerance	Resets the absolute voltage error tolerance of the program.	1.0e-06	V	Generally, set to 6 to 8 orders of magnitude smaller than the largest voltage signal in the circuit.
CHGTOL	Charge error tolerance	Resets the charge tolerance in coulombs.	1.0e-14	C	Do not change default.
RELTOL	Relative error tolerance	Resets the relative error tolerance of the simulation, which is the universal accuracy control. The value can significantly affect the convergence of the solution and the simulation speed. Value must be between 1 and 0.	0.001	—	Use typical values between 1.0e-06 and 0.01.
GMIN	Minimum conductance	Resets the minimum conductance used in any circuit branch. Cannot be zero. Increasing this may positively improve the convergence of the solution; however, it will also negatively affect simulation accuracy.	1.0e-12	mho	Do not change default.
PIVREL	Minimum acceptable ratio of pivot	Resets the relative value between the largest column entry in the matrix and an acceptable pivot value. Value must be between 1 and 0.	0.001	—	Do not change default.

Code	Option Name	Description	Default	Unit	Recommendation
PIVTOL	Minimum acceptable pivot	Resets the absolute minimum value for a matrix entry to be accepted as a pivot.	1.0e-13	—	Do not change default.
TEMP	Operating temperature	Resets the temperature at which the entire circuit will be simulated. Setting in the Analysis Parameters dialog box will override.	27	°C	—
RSHUNT	Shunt resistance from analog nodes to ground	Inserts resistance to ground at all analog nodes in the circuit. Reducing value reduces simulation accuracy.	1.0+e12	Ω	Should be set to some very high resistance, say 1e+12Ω If you get a “No DC path to ground” or a “Matrix is nearly singular” error message, try decreasing RSHUNT to 1e+9Ω or 1e+6Ω.
RAMPTIME	Transient analysis supply ramping time	Ramps independent sources, capacitor and inductor initial conditions from zero to their final values during the time period specified.	0	s	—
CONVSTEP	Fractional step allowed by code model inputs between iterations	Controls automatic convergence assistance by establishing a relative step size limit in solving for the DC operating point.	0.25	—	—
CONVABSSTEP	Absolute step allowed by code model inputs between iterations	Controls automatic convergence assistance by establishing an absolute step size limit in solving for the DC operating point.	0.1	—	—

Code	Option Name	Description	Default	Unit	Recommendation
CONVLIMIT	Enable convergence assistance on code models	Enables/disables a convergence algorithm used in some built-in component models.	On	—	—
ACCT	Print simulation statistics	Turns on/off display of statistical data on simulation-related information. Data may be useful for debugging simulation-related problems. Data appears in the Grapher dialog box.	Off	—	—

DC Tab

Code	Option Name	Description	Default	Unit	Recommendation
ITL1	DC iteration limit	Resets the upper bound limit to the number of Newton-Raphson iterations during a DC operating point analysis.	100	—	If you receive the error message “No convergence in DC analysis”, increase the ITL1 value to 500 or 1000 and rerun the analysis.
ITL2	DC transfer curve iteration limit	Resets the DC transfer curve iteration limit.	50	—	—
ITL6	Steps in source stepping algorithm	Sets the number of steps in the Static Source Stepping algorithm. The value only has an effect only if the DYNAMICDC option is not ON. The Static Source Stepping algorithm helps find a solution during a DC operating point analysis. Refer to the *DC Convergence Assistance Algorithms* section of Chapter 8, *Simulation*, for more information.	10	—	Increase if increasing ITL1 option does not help convergence.

Code	Option Name	Description	Default	Unit	Recommendation
GMINSTEPS	Number of Gmin steps	Sets the number of steps in the Static Diagonal GMIN stepping algorithm. The value only has an effect only if the DYNAMICDC option is not ON. The Static Diagonal GMIN Stepping algorithm helps find a solution during a DC operating point analysis. Refer to the *DC Convergence Assistance Algorithms* section of Chapter 8, *Simulation*, for more information.	10	—	Increase if increasing ITL1 option does not help convergence.
DYNAMICDCFACTOR	Dynamic DC Factor	Sets the initial and upper bound factor value by which the DC sources and Diagonal GMIN values are modified during the Dynamic DC Stepping algorithms. Note that this value only has an effect if the DYNAMICDC option is ON. The Dynamic Source and Diagonal GMIN Stepping algorithms help find a solution during a DC operating point analysis. Refer to the *DC Convergence Assistance Algorithms* section of Chapter 8, *Simulation*, for more information.	10	—	Increase if increasing ITL1 option does not help convergence.

Code	Option Name	Description	Default	Unit	Recommendation
DYNAMICDC	Dynamic DC Stepping	Selects the Dynamic DC Stepping algorithms for convergence assistance if set to ON. Otherwise, Static DC Stepping algorithms are used.	ON	—	Set this option to ON, as the Dynamic Stepping algorithms are generally more powerful than the Static Stepping algorithms.
NOOPITER	Go directly to Gmin stepping	If set, the basic DC algorithm will not be attempted; only the Static or Dynamic stepping algorithms will attempt to find the operating point.	OFF	—	—

Transient Tab

Code	Option Name	Description	Default	Unit	Recommendation
ITL4	Upper transient iteration limit	Resets the upper bound limit to the number of Newton-Raphson iterations at each transient time point. Increasing the value may slow down transient simulation time. Decreasing the value increases the chance of in-convergence.	100	—	If you receive the error message “Time step too small” or “No convergence in transient analysis”, increase the ITL4 value to 150 and rerun the analysis.
MAXORD	Maximum integration order	Sets the maximum order for integration when GEAR chosen as transient analysis integration method. Must be between 2 and 6. Using a higher order theoretically leads to more accurate results, but slows down simulation.	2	—	Use the default value for most circuit simulation.
TRTOL	Truncation error overestima-tion factor	Resets transient error tolerance. Only used in the local truncation error criterion.	7	—	Use default value

Code	Option Name	Description	Default	Unit	Recommendation
METHOD	Integration method	Selects for transient analysis. Default provides faster simulations with same numerical accuracy, but can produce unintended results.	TRAPEZOI-DAL	—	Use GEAR (gear integration method) if unwanted numerical oscillations occur during simulation or if circuit contains ideal switches. Use default if circuit operates in oscillation mode, for example, oscillator circuits. Be aware that Gear integration may overdamp results.
ADERROR	Maximum Analog-to-Digital Interface Error	Sets the maximum time that elapses between the time an analog node passes an A/D bridge threshold and the time that this bridge changes its digital output state.	0.1n	s	Small values will increase accuracy at the expense of decreased simulation speed. Use as needed.

Device Tab

Code	Option Name	Description	Default	Unit	Recommendation
TNOM	Nominal temperature	Resets the normal temperature at which model parameters are measured and calculated.	27	°C	Do not change unless you want your circuit to match data book specifications that were extracted at a temperature other than 300.15 °K.
BYPASS	Allow bypass of unchanging elements	Turns off/on the device bypass scheme for nonlinear model evaluation. Turning off may increase simulation time	Off	—	Do not change default.
DEFAD	Default MOSFET area of drain	Resets the value for MOS drain diffusion area.	0	m^2	Use default value unless you know how to specify a value from a MOS device datasheet.
DEFAS	Default MOSFET area of source	Resets the value for MOS source diffusion area.	0	m^2	Use default value unless you know how to specify a value from a MOS device datasheet.
DEFL	Default MOSFET length	Resets the value for MOS channel length.	0.0001	m	Use default value unless you know how to specify a value from a MOS device datasheet.
DEFW	Default MOSFET width	Resets the value for MOS channel width	0.0001	m	Use default value unless you know how to specify a value from a MOS device datasheet.

Code	Option Name	Description	Default	Unit	Recommendation
TRYTOCOMPACT	Try compaction for LTRA lines	Applicable only to lossy transmission line component. When option turned on, Multisim tries to reduce data storage and memory usage needed for transient simulation of circuits containing lossy transmission lines.	Off	—	—
OLDLIMIT	Use SPICE2 MOSFET limiting	—	Off	—	—

Advanced Tab

Code	Option Name	Description	Default	Unit	Recommendation
AUTOPARTIAL	Use auto-partial computation for all models	—	Off	—	—
BADMOS3	Use old mos3 model (discontinuous with respect to kappa)	—	Off	—	—
KEEPOPINFO	Record operating point for each small-signal analysis	Retains the operating point information whether an AC, Distortion, or Pole-Zero analysis is run.	Off	—	Particularly useful if the circuit is large and you do not want to run a redundant ".OP" analysis.
MAXEVTITER	Maximum event iterations at analysis point		0	—	—
MAXOPALTER	Maximum analog/event alternations in DCOP		0	—	—

Code	Option Name	Description	Default	Unit	Recommendation
MINBREAK	Minimum time between breakpoints		0	—	—
NOOPALTER	Do not do analog/event alternation in DCOP		Off	—	—

11

Postprocessor

This chapter explains how to use the **Postprocessor** to mathematically manipulate the results of simulation obtained through analyses in different ways.

To use the **Postprocessor**, you must have performed at least one analysis on your circuit. This chapter assumes that you are familiar with the analyses offered by Multisim, and the **Grapher** that displays analysis results. Refer to Chapter 10, *Analyses* for more information.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Introduction to the Postprocessor

Use the **Postprocessor** to manipulate the output from analyses performed on a circuit and plot the results on a graph or chart. The plotted results are referred to as "traces". Types of mathematical operations that can be performed on analysis results include: arithmetic, trigonometric, exponential, logarithmic, complex, vector, and logic.

The following illustrate possible uses of the **Postprocessor**:

- Divide the output curve by the input curve obtained from a transient analysis, and observe the results.
- Multiply a voltage by a current to observe circuit power.
- Assess the differences caused by minor changes to your circuit. For example, run an analysis on a circuit, then change one condition of the circuit (such as changing the input voltage of the component's value) and run the analysis again. Subtract one set of results from the other to show the effect of the circuit modification.

Using the Postprocessor

The **Postprocessor** calculates the results of equations and plots these results as "traces" on graphs and charts. To use the **Postprocessor**, you build the equations by combining the variables from previous circuit analysis results with mathematical functions.

To build equations for the **Postprocessor**, you must have performed at least one analysis. When you perform an analysis on a circuit, the results appear in the **Grapher** and are stored for use by the **Postprocessor**. Refer to Chapter 10, *Analyses* for information about performing analyses.

Note Equations created in the **Postprocessor** will be saved with Multisim allowing them to be available in the **Postprocessor** the next time you open Multisim. Your expressions can be used with other analysis results as long as the expression variables are labeled the same.

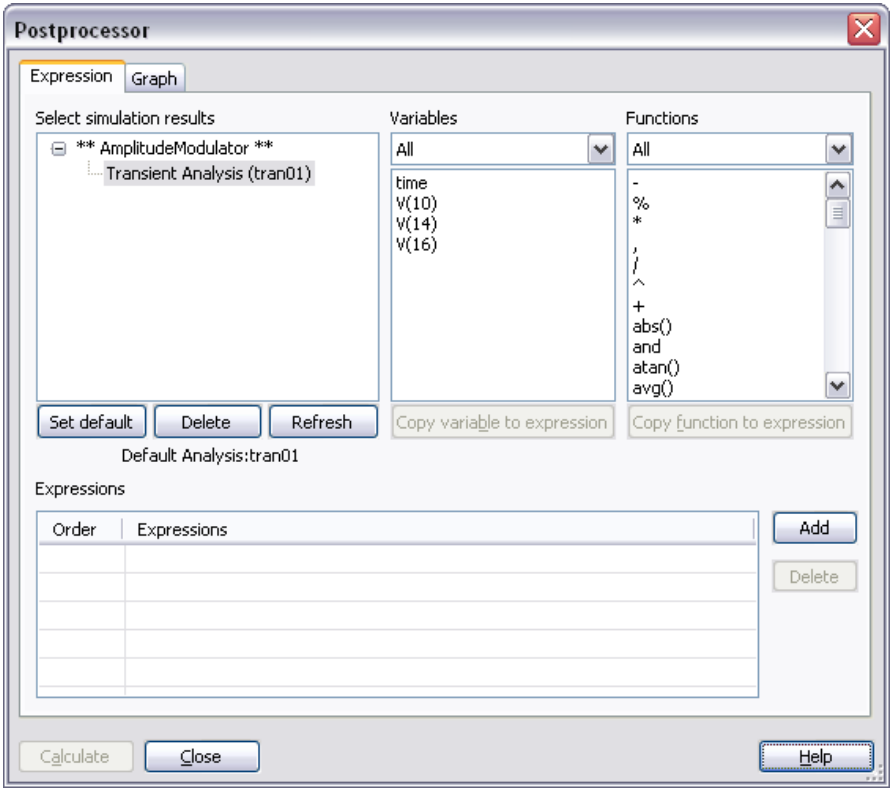
Basic Steps

To construct an equation from which a trace will be plotted, you select variables and mathematical functions.

Note You must have done an analysis on the circuit you wish to use with the postprocessor in the current Multisim session so that the current simulation results are available.

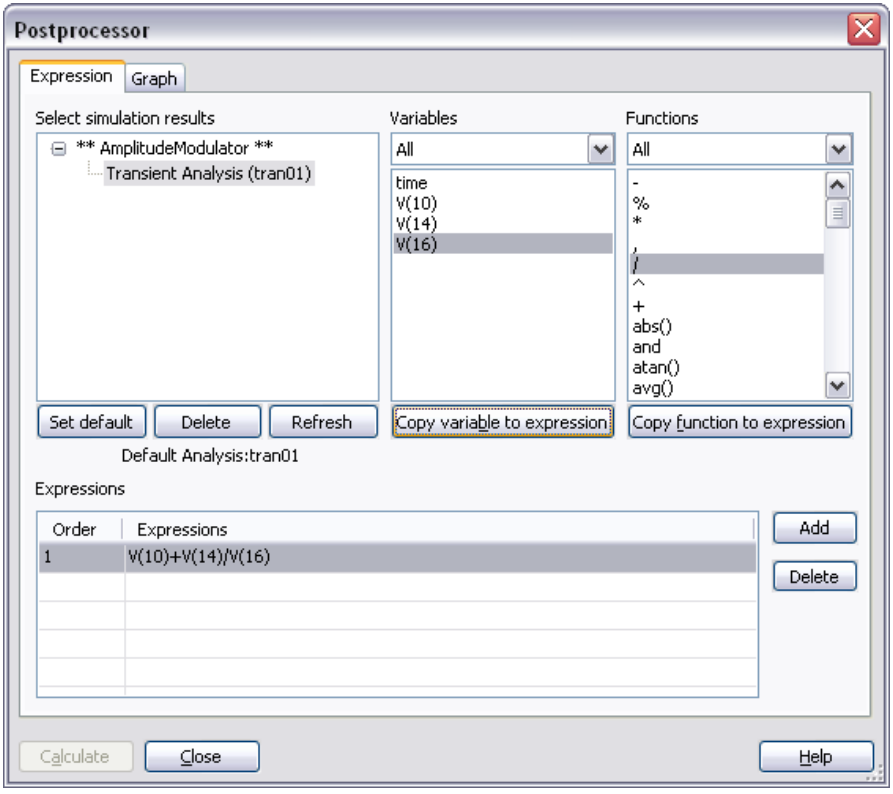
Complete the following steps to build an expression:

1. Click the **Postprocessor** button on the **Main** toolbar or select **Simulate»Postprocessor**. The **Postprocessor** dialog box appears with the following information:
   - **Select simulation results**—This list contains names of the circuits on which analyses have been performed in this session. For example, Amplitude Modulator in the figure below. Below each circuit name is a list of the specific analyses performed.
   - **Variables**—This lists the variables that resulted from the analysis selected in **Select simulation results**.
   - **Functions**—This lists the mathematical functions available for use in your expression.
   - **Expressions**—This is where you build your **Postprocessor** expression. Expressions from earlier sessions may be here. To remove, highlight the expression and click **Delete**.



2. Click on the **Expression** tab. The **Select Simulation Results** section lists Multisim files and each analysis performed within the file. Each analysis is identified with a name followed by a code in brackets. That code will be used to identify the variables from that analysis when the trace is plotted. Click on the analysis name; the variables that resulted from that analysis appear in the **Variables** list.
3. From the **Variables** list, select the variable you want to include in the equation being used to define the trace, and click **Copy Variable to Expression**. The variable appears in the **Expressions** section, prepended with the code of the analysis from which it is drawn, unless the selected analysis is the default analysis. Refer to the *Using the Default Analysis* section for more information.

Note The **Variables** list contains the variables selected in the **Selected Variables for Analysis** list in the analyses **Output** tabs. To filter the **Variables** list to show only certain variables, choose from the drop-down list, for example, All Voltages.

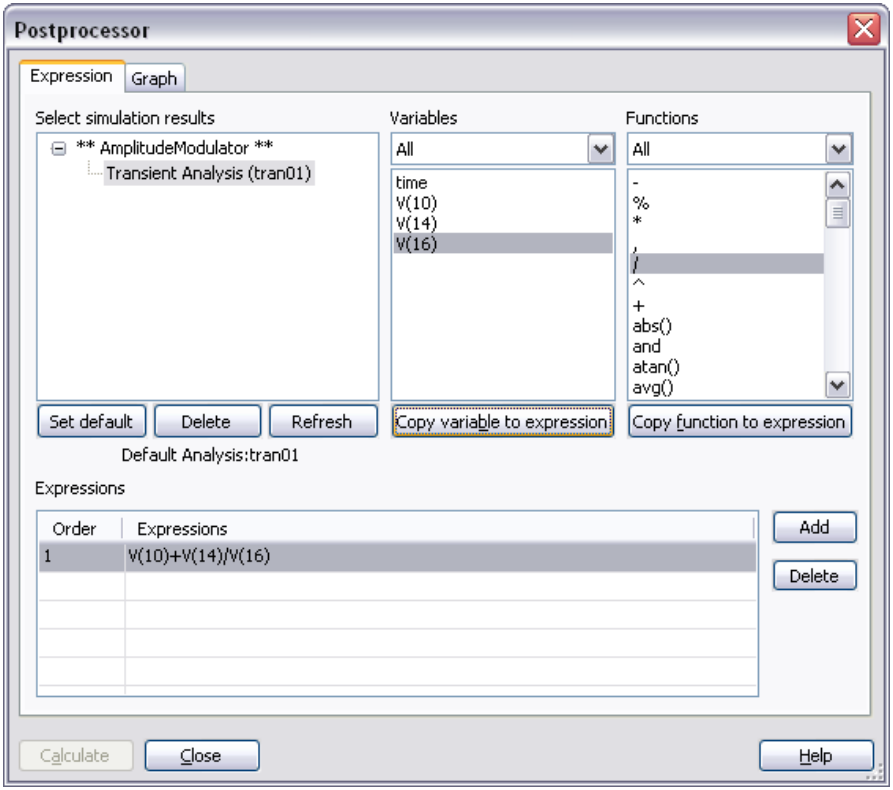


4. From the **Functions** list, select the mathematical function you want included in the equation for this trace, and click **Copy Function to Expression**. To filter the **Functions** list to show only certain mathematical functions, choose from the drop-down list of options. Refer to the *Available Postprocessor Functions* section for more information.

Note It is possible to manually type or modify a trace's equation although you should exercise caution if doing so.

5. Continue to choose analyses, variables and functions until the equation is complete.

6. When complete, click **Add** or press Enter to save your equation in the **Expressions** section, as in the example below.

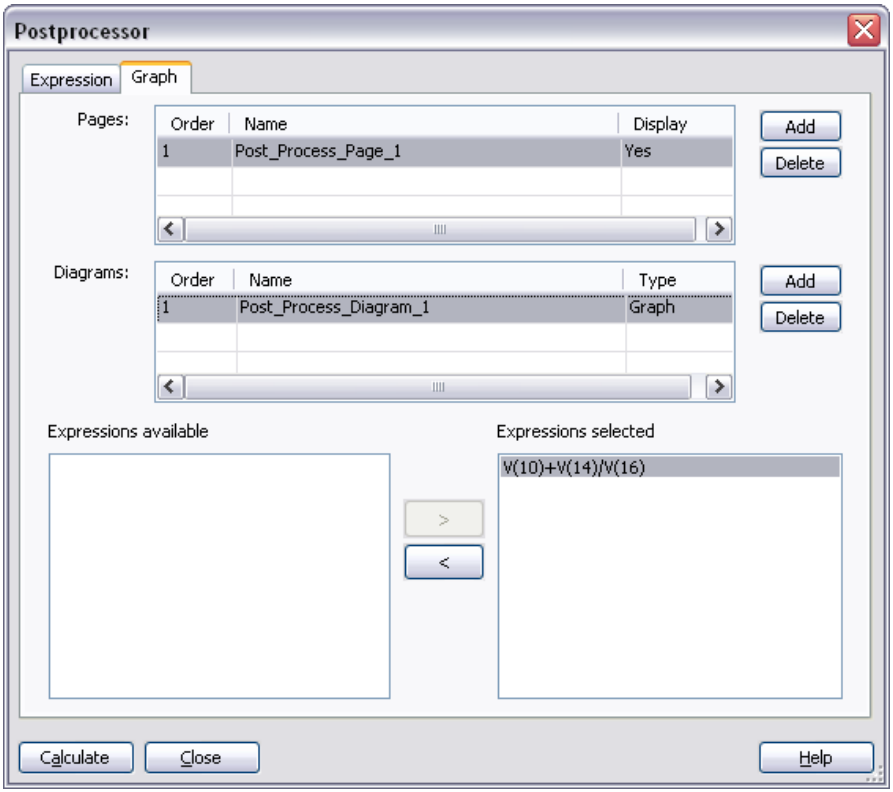


7. Repeat above steps to add more equations.

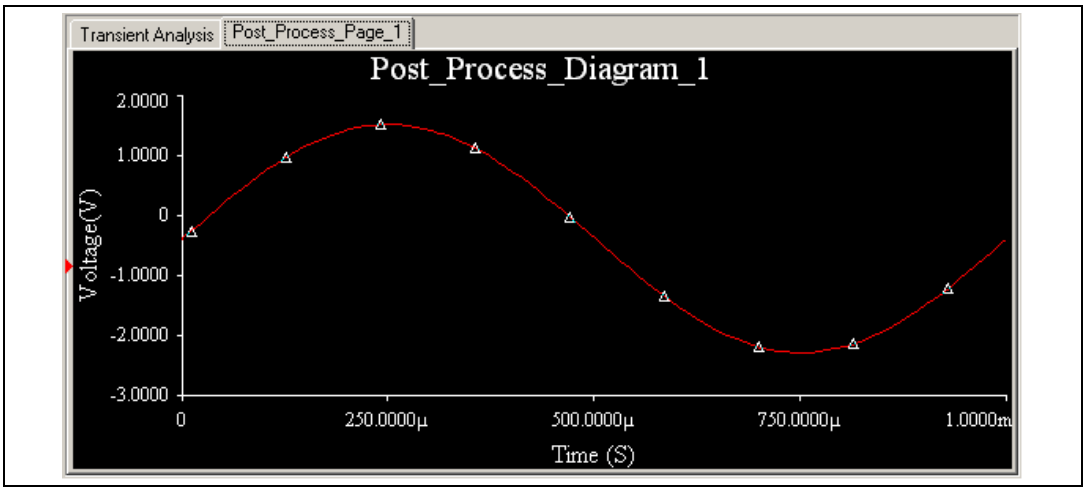
Complete the following steps to view the results of the built equation:

1. Select the **Graph** tab in the **Postprocessor** dialog box.
2. Click the **Add** button to the right of the **Pages** section. A default name will appear in the **Name** column of the **Pages** section. This is the name of the tab that will display the results in the **Grapher**. Change the default name if necessary.
3. Click the **Add** button to the right of the **Diagrams** section. A default name will appear in the **Name** column of the **Diagrams** section. This is the title of the diagram that will be displayed in the **Grapher**. Change the default name if necessary.
4. Click in the **Type** column of the **Diagram** section and choose either **Graph** or **Chart** from the drop-down list that appears.

5. In the **Expressions Available** section, select the equation you want to view.
6. Click the **>** button to move the equation to the **Expressions Selected** field. The dialog box will resemble the following:



7. Click the **Calculate** button to open the **Grapher** and view the results.



The results of the **Postprocessor** plotting the trace appear in the **Grapher**, in tabs with the names you specified, along with the results of the analyses previously performed. Results, including errors, are also recorded in the **Simulation Error Log/Audit Trail**. Refer to the *Simulation Error Log/Audit Trail* section of Chapter 8, *Simulation*, for more information.

Using the Default Analysis

The equation you build using the **Postprocessor** contains variables that are prepended with their analysis' code. To simplify the equation and the trace displayed on the graph, you can set one of the analyses to be the default analysis.

The **Select Simulation Results** list always contains one analysis defined as the default. The default is the analysis that, in the absence of any other indication, the **Postprocessor** uses for calculations. Variables from the default analysis do not have identifying prefixes in the equation or when the trace is plotted.

The default analysis is identified below the **Select Simulation Results** section in the **Expression** tab.

Complete the following steps to change the default analysis:

1. Click on the desired analysis in the **Select Simulation Results** section.
2. Click **Set Default**. The **Default Analysis** field displays your choice.

Creating Multiple Traces

Complete the following steps to add another trace to a **Grapher** page:

1. Select the **Graph** tab from the **Postprocessor** dialog box.
2. In the **Expressions Available** section, select another equation you want to view.
3. Click the **>** button to move the equation to the **Expressions Selected** field.
4. Click the **Calculate** button to open the **Grapher** and view the results.

Working with Pages, Traces, Graphs and Charts

Complete the following steps to add another page (tab) to the **Grapher** to display traces:

1. Select the **Graph** tab in the **Postprocessor** dialog box.
2. Click the **Add** button in the **Pages** section. A default name will appear in the **Name** column under the first page name. Change the default name if necessary.
3. Click the **Add** button in the **Diagrams** section. A default name will appear in the **Name** column under the first diagram name. Change the default name if necessary.
4. Click in the **Type** column of the **Diagram** section and choose either **Graph** or **Chart** from the drop-down list that appears.
5. In the **Expressions Available** section, select the equation you want to view.
6. Click the **>** button to move the equation to the **Expressions Selected** field.
7. Click the **Calculate** button to open the **Grapher** and view the results.

Complete the following steps to change between a graph or chart:

1. Select the **Graph** tab in the **Postprocessor** dialog box.
2. Click in the **Type** column of the **Diagram** section and choose either Graph or Chart from the drop-down list that appears.

Note Each chart or graph appears on the page in the **Grapher** to which it has been assigned.

Complete the following steps to remove a trace:

1. Select the **Graph** tab and highlight the expression to be removed in the **Expressions Selected** field.
2. Click the **<** button to move the expression to the **Expressions Available** field.

To delete a page, select it in the **Pages** section of the **Graph** tab and click the **Delete** button to the right of the **Pages** section.

Postprocessor Variables

The variables that appear in the **Analysis Variables** list of the **Postprocessor** are based on the selected analysis. They can include any or all of the following:

- All variables
- All Voltages
- Voltages except submodules
- Voltages only at static probes
- All Currents
- Currents except submodules
- Currents only at static probes
- All Powers
- Powers except submodules.

Available Postprocessor Functions

The functions you can apply to the **Postprocessor** variables are detailed in the following sections (symbol followed by description).

Algebraic Functions

+—plus
-—minus
*****—times
/—divided by
^—to the power of
%—modulus
,—complex 3,4 = 3 + j (4)
abs(X)—absolute value
sqrt(X)—square root

Trigonometric Functions

sgn(X)—1(if x>0), 0(if x=0), -1(if x<0)
sin(X)—Trigonometric sine (argument in radians)
cos(X)—Trigonometric cosine (argument in radians)
tan(X)—Trigonometric tangent (argument in radians)
atan(X)—Trigonometric inverse tangent

Relational Functions

gt—Greater than
lt—Less than
ge—Greater than or equal to
le—Less than or equal to
ne—Not equal to
eq—Equal to

Logical Functions

and—And
or—Or
not—Not

Exponential Functions

db(X)—Decibels 20 $\log_{10}$(mag(X))
log(X)—Logarithm (base 10)
ln(X)—Natural logarithm (base e)
exp(X)—Exponential e to the vector power

Complex Functions

j(X)—Complex i (sqrt(-1)) times X
real(X)—Complex real component of X
imag(X)—Complex imaginary part of X
vi(X)—Complex vi(X) = image (v(X))
vr(X)—Complex vr(X) = real (v(X))

Vector Functions

avg(X)—Running average of the vector X where:

avgx(X, d)—Running average of the vector X over d where:

if $x_i - d = x_0$, otherwise:

$$\text{avg}(\mathbf{X}(x_i))_i = \frac{\int_{x_o}^{x_i} \mathbf{X}(x)dx}{x_i - x_o}$$

$$\text{avgx}(\mathbf{X}(x_i))_i = \frac{\int_{x_i - d}^{x_i} \mathbf{X}(x)dx}{x_i - d}$$

$$\text{avgx}(\mathbf{X}(x_i))_i = \frac{\int_{x_o}^{x_i} \mathbf{X}(x)dx}{x_i - x_o}$$

deriv(X)—Vector derivative of X - uses numeric differentiation by interpolating a polynomial and may not produce satisfactory results, particularly with iterated differentiation. Only calculates the derivative with respect to the real component of the vector's scale

envmax(X, n)—Upper envelope of the vector X where n is the number of points on either side of a peak that must be less than the value for a peak to be identified.

envmin(X, n)—Lower envelope of the vector X where n is the number of points on either side of a valley that must be greater than the value for a valley to be identified.

grpdelay(X)—Group delay of vector X in seconds, where:

$$\text{grpdelay}(\mathbf{X}(freq))_i = -\frac{1}{360}\left[\frac{d[\text{ph}(\mathbf{X}(freq))]}{dfreq}\right]_{freq_i} = -\frac{1}{360}\text{deriv}[\text{ph}(\mathbf{X}(freq))]_{freq_i}$$

integral(X)—Running integral of vector X, where:

$$\text{integral}(\mathbf{X}(x_i))_i = \int_{x_o}^{x_i} \mathbf{X}(x)dx$$

mag(X)—Vector magnitude
ph(X)—Vector phase
norm(X)—Vector X normalized to 1 where:

$$\text{norm}(\mathbf{X}) = \frac{\mathbf{X}}{\max(\text{abs}(\mathbf{X}))}$$

rms(X)—Running RMS average of vector X where:

$$\text{rms}(\mathbf{X}(x_i))_i = \sqrt{\frac{\int_{x_o}^{x_i} |\mathbf{X}(x)|^2 dx}{x_i - x_o}}$$

rnd(X)—Vector random
mean(X)—Vector results in a scalar (a length 1 vector) that is the mean of the elements of the vector
Vector(n)—Vector results in a vector of length n, with elements 0, 1, ... n-1. If n is a vector than just the first element is taken, and if it isn't an image then the floor of the magnitude is used
length(X)—Vector length of vector X
max(X)—Vector maximum value from X
min(X)—Vector minimum value from X
vm(X)—Vector vm(x) = mag(v(X))
vp(X)—Vector vp(x) = ph(v(X))

Constant Functions

yes—Yes
true—True
no—No
false—False
pi—Pi
e—Natural logarithm base
c—Speed of light in vacuum
i—Square root of –1
kelvin—Absolute zero in Celsius
echarge—Fundamental charge
boltz—Boltzman's constant
planck—Planck's constant

12

Reports

This chapter details the various reports that are available in Multisim.

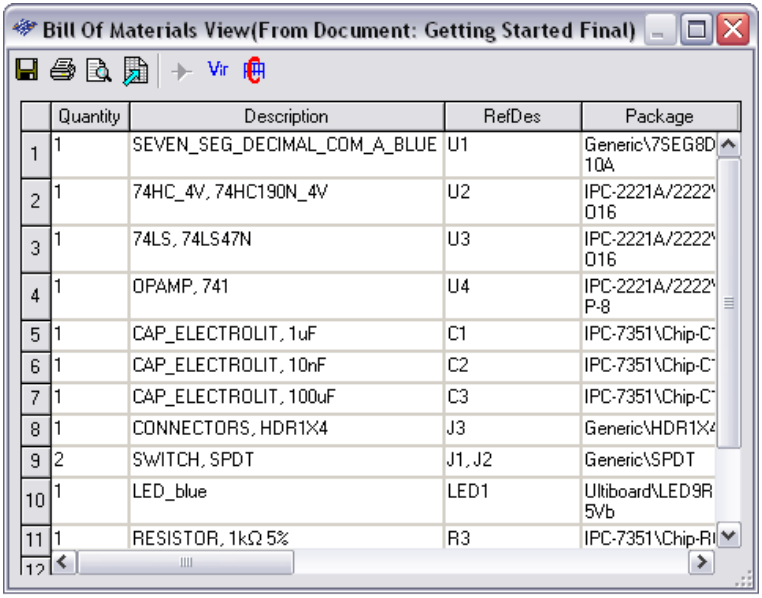
Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Bill of Materials

A **Bill of Materials** (BOM) lists the components used in your design and provides a summary of the components needed to manufacture the circuit board. You can also select which fields to view and print on your BOM.

The **Bill of Materials** report includes each component's:

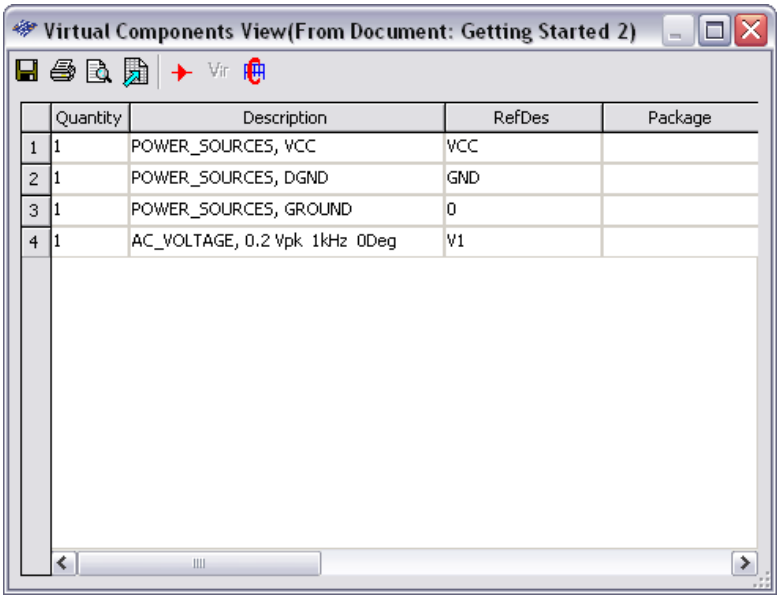
- Quantity
- Description, including its type (for example, resistor) and value (for example, 5.1 kohm)
- RefDes
- Package or footprint
- Type



	Quantity	Description	RefDes	Package
1	1	SEVEN_SEG_DECIMAL_COM_A_BLUE	U1	Generic\7SEG8D 10A
2	1	74HC_4V, 74HC190N_4V	U2	IPC-2221A/2222 O16
3	1	74LS, 74LS47N	U3	IPC-2221A/2222 O16
4	1	OPAMP, 741	U4	IPC-2221A/2222 P-8
5	1	CAP_ELECTROLIT, 1uF	C1	IPC-7351\Chip-C
6	1	CAP_ELECTROLIT, 10nF	C2	IPC-7351\Chip-C
7	1	CAP_ELECTROLIT, 100uF	C3	IPC-7351\Chip-C
8	1	CONNECTORS, HDR1X4	J3	Generic\HDR1X4
9	2	SWITCH, SPDT	J1, J2	Generic\SPDT
10	1	LED_blue	LED1	Ultiboard\LED9R 5Vb
11	1	RESISTOR, 1kΩ 5%	R3	IPC-7351\Chip-R
12				

The **Bill of Materials** can also show all user fields and their values. Refer to the *Editing User Fields* section of Chapter 6, *Component Editing*, for information about defining and completing user fields.

Note Because a BOM is primarily intended to assist in procurement and manufacturing, it includes only “real” parts. Therefore, it excludes parts that are not real or able to be purchased, such as sources or virtual components. If desired, you may also view a list of the virtual components in a circuit, as in the example shown in the figure below.



Using the BOM Report

Complete the following steps to create a **Bill of Materials** (BOM) for your circuit:

1. Have the desired circuit open in your workspace.
2. Select **Reports»Bill of Materials**. The **Bill of Materials View** dialog box displays.

Note If there are multiple variants in the circuit, the **Variants Filter** dialog box will display before the report dialog. Refer to the *Variants Filter Dialog Box* section for more information.

3. Optionally, adjust the displayed information using the buttons detailed below.

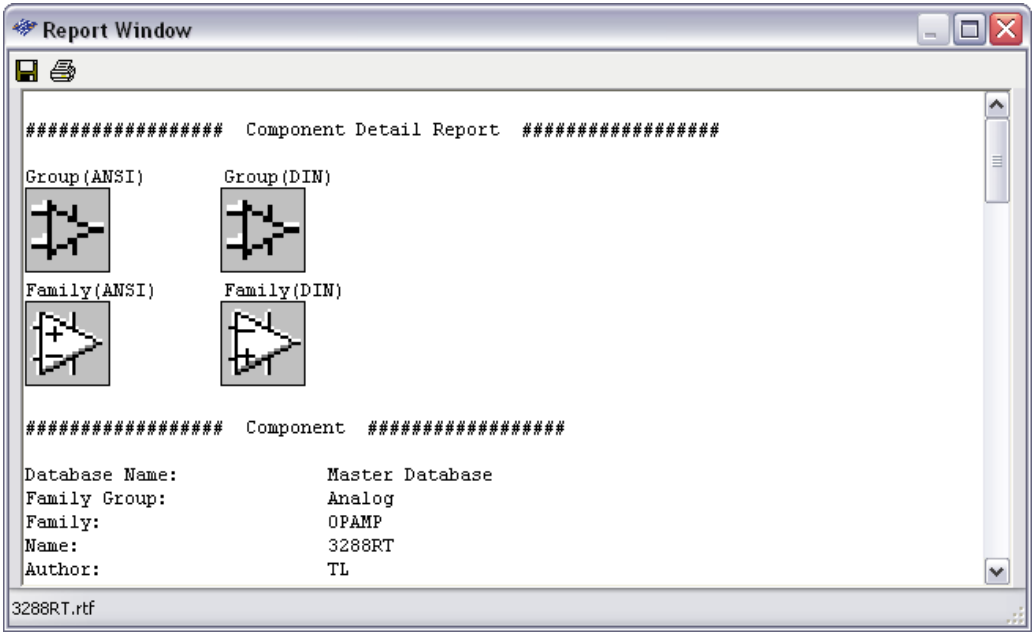
Button	Description
	Show Real Components button. Click to return to the **Bill of Materials View** dialog box from either the **Select Columns** dialog box or the **Virtual Components View** dialog box.
	Show Virtual Components button. Click to display the **Virtual Components View** dialog box.
	Select Columns button. Click to display the **Select Columns** dialog box. Select or deselect the columns you wish to view. Note that the buttons on the left side of the toolbar are greyed-out (disabled) when this view is selected.

4. Optionally, click on a column (**Quantity**, **Description**, etc.) to sort the data in ascending order by that column. Click again to sort by descending order.
5. Use the buttons detailed below as required to produce your report.

Button	Description
	Save to a Text File button. Click to save the data (from either the **Bill of Materials View** or the **Virtual Components View**) to a text file. A standard Windows save dialog box appears. Choose the desired filepath and click **Save**.
	Print button. Click to display a standard Windows print dialog box. Choose the desired print options and click **OK**.
	Print Preview button. Click to display the **Print Preview** dialog box.
	Export to MS Excel button. Click to open a Microsoft® Excel spreadsheet with the data from the dialog box displayed. (You must have Excel installed to use this function.)

Component Detail Report

You can produce a **Component Detail Report** showing all information stored in the Multisim database for a particular component.



Using the Component Detail Report

Complete the following steps to produce a **Component Detail Report** showing detailed information about a specific component:

1. Select **Reports»Component Detail Report**. The **Select a Component to Print** dialog box displays.
2. Select a specific **Database**, **Group**, **Family** and **Component** in the dialog box.
3. Click the **Detail Report** button to display the report window.
4. Scroll through the information as necessary using the scrollbar on the right of the dialog box.

5. Use the following buttons on the dialog box as required:

Button	Description
	Save button. Click to save the data to a text file. A standard Windows save dialog box appears. Choose the desired filepath and click **Save**.
	Print button. Click to print the information in the report. A standard Windows print dialog box appears. Choose the desired print options and click **OK**.

Note This report can also be accessed via the **Detail Report** button in the **Components** tab of the **Database Manager** dialog box.

Netlist Report

The **Netlist Report** provides the following circuit connectivity information for each component:

- **Net** (net name)
- **Page** (filename)
- **Pin** (logical pin name).

Using the Netlist Report

Complete the following steps to produce a netlist:

1. Have the desired circuit open in your workspace.
2. Select **Reports»Netlist Report**. The **Netlist Report** dialog box displays.

Note If there are multiple variants in the circuit, the **Variants Filter** dialog box will display before the report dialog. Refer to the *Variants Filter Dialog Box* section for more information.

3. Optionally, click on a column (for example, **Net**, **Page**) to sort the data on the dialog box in ascending order by that column. Click again to sort by descending order.
   - **Net**—The net to which the **Component Pin** is connected.
   - **Page**—Page (filename) where the component is located or the root page if the component is part of a subcircuit, hierarchical block or multi-page.

- **Component**—Component's RefDes and subcircuit, hierarchical block or multi-page name.
- **Pin**—Logical pin name.

4. Use the buttons detailed below as required to produce your report.

Button	Description
	Save to a Text File button. Click to save the data in the dialog box to a text file. A standard Windows save dialog box appears. Choose the desired filepath and click **Save**.
	Print button. Click to display a standard Windows print dialog box. Choose the desired print options and click **OK**.
	Print Preview button. Click to display the **Print Preview** dialog box.
	Export to MS Excel button. Click to open a Microsoft® Excel spreadsheet with the data from the dialog box displayed. (You must have Excel installed to use this function).

Schematic Statistics Report

The **Schematic Statistics Report** lists the quantity of the following in your circuit:

- Components—Total number of components. (Real components + virtual components).
- Real components—Components that can be purchased.
- Virtual components—Components that cannot be purchased.
- Gates—Total number of gates used in the design.
- Nets—Total number of connections between pins.
- Pins in nets.
- Unconnected pins.
- Total pins—Pins in nets + unconnected pins.
- Pages.
- Hierarchical blocks—Total number of hierarchical blocks, unique or otherwise. Instances of hierarchical blocks equals the number of copies of the same hierarchical block.

- Unique hierarchical blocks—Total number of unique hierarchical blocks.
- Subcircuits—Total number of subcircuits, unique or otherwise.
- Unique subcircuits.

Using the Schematic Statistics Report

Complete the following steps to produce a schematic statistics report:

1. Have the desired circuit open in your workspace.
2. Select **Reports»Schematic Statistics**. The **Schematic Statistics Report** dialog box displays.

Note If there are multiple variants in the circuit, the **Variants Filter** dialog box will display before the report dialog. Refer to the *Variants Filter Dialog Box* section for more information.

3. Optionally, click on a column (**Name**, **Quantity**) to sort the data on the dialog box in ascending order by that column. Click again to sort by descending order.
4. Use the buttons detailed below as required to produce your report.

Button	Description
	Save to a Text File button. Click to save the data in the dialog box to a text file. A standard Windows save dialog box appears. Choose the desired filepath and click **Save**.
	Print button. Click to display a standard Windows print dialog box. Choose the desired print options and click **OK**.
	Print Preview button. Click to display the **Print Preview** dialog box.
	Export to MS Excel button. Click to open a Microsoft® Excel spreadsheet with the data from the dialog box displayed. (You must have Excel installed to use this function).

Spare Gates Report

The **Spare Gates Report** lists the unused sections of multi-section components in a circuit. By running the **Spare Gates Report**, you can quickly see which components still have available gates.

For example, in a QUAD 2-INPUT NAND you may not have used all four of the available NAND gates (sections).

Using the Spare Gates Report

Complete the following steps to produce a **Spare Gates Report**:

1. Have the desired circuit open in your workspace.
2. Select **Reports»Spare Gates Report**. The **Spare Gates Report** dialog box displays.

Note If there are multiple variants in the circuit, the **Variants Filter** dialog box will display before the report dialog. Refer to the *Variants Filter Dialog Box* section for more information.

3. Optionally, click on a column to sort the data on the dialog box in ascending order by that column. Click again to sort by descending order.

   The columns contain the following information:

   - **RefDes**—The reference designator of the component that contains the spare gates.
   - **Part**—The type of component.
   - **Section(s) available**—The sections in the component that are still unplaced. Each section corresponds to one gate.
4. Use the buttons detailed below as required to produce your report.

Button	Description
	Save to a Text File button. Click to save the data in the dialog box to a text file. A standard Windows save dialog box appears. Choose the desired filepath and click **Save**.
	Print button. Click to display a standard Windows print dialog box. Choose the desired print options and click **OK**.

Button	Description
	Print Preview button. Click to display the **Print Preview** dialog box.
	Export to MS Excel button. Click to open a Microsoft® Excel spreadsheet with the data from the dialog box displayed. (You must have Excel installed to use this function).

Model Data Report

The **Model Data Report** gives you the model information for a selected component.

Using the Model Data Report

Complete the following steps to generate a **Model Data Report**:

1. Select **Place»Component** to display the **Select a Component** dialog box.
2. Use the **Database**, **Group**, **Family** and **Component** lists to navigate to, and highlight, the component for which you wish to generate the report.
3. Click on **View Model**. The **Model Data Report** displays.

Note You can also click the **View Model** button from the **Select a Component to Print** dialog box, which appears when you select **Reports»Component Detail Report**. Refer to the *Component Detail Report* section for information about the **Component Detail Report**.

4. Use the following buttons on the dialog box as required:

Button	Description
	Save button. Click to save the data to a text file. A standard Windows save dialog box appears. Choose the desired filepath and click **Save**.
	Print button. Click to print the information in the report. A standard Windows print dialog box appears. Choose the desired print options and click **OK**.

Cross Reference Report

The **Cross Reference Report** provides a detailed list of all components in the active circuit.

Using the Cross Reference Report

Complete the following steps to produce a **Cross Reference** report:

1. Have the desired circuit open in your workspace.
2. Select **Reports»Cross Reference Report**. The **Cross Reference Report** dialog box displays.

Note If there are multiple variants in the circuit, the **Variants Filter** dialog box will display before the report dialog. Refer to the *Variants Filter Dialog Box* section for more information.

3. Optionally, click on a column to sort the data on the dialog box in ascending order by that column. Click again to sort by descending order.
4. Use the buttons detailed below as required to produce your report.

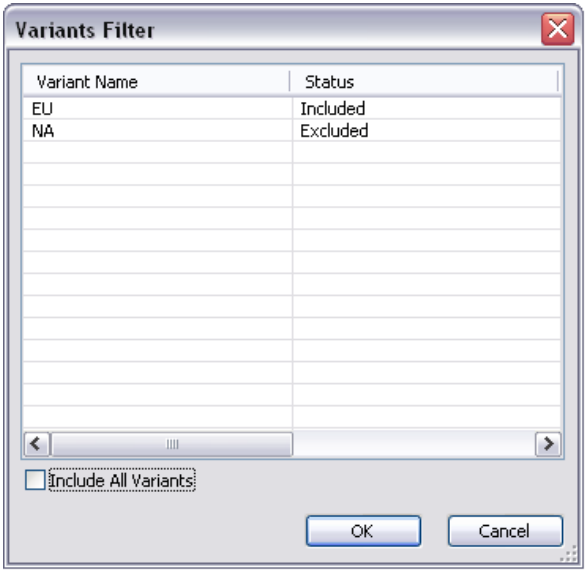
Button	Description
	Save to a Text File button. Click to save the data in the dialog box to a text file. A standard Windows save dialog box appears. Choose the desired filepath and click **Save**.
	Print button. Click to display a standard Windows print dialog box. Choose the desired print options and click **OK**.
	Print Preview button. Click to display the **Print Preview** dialog box.
	Export to MS Excel button. Click to open a Microsoft® Excel spreadsheet with the data from the dialog box displayed. (You must have Excel installed to use this function).

Variants Filter Dialog Box

When there is more than one variant present in a circuit, the **Variants Filter** dialog box displays when you run the following reports:

- **Bill of Materials**—Refer to the *Bill of Materials* section for information.
- **Netlist Report**—Refer to the *Netlist Report* section for information.
- **Schematic Statistics**—Refer to the *Schematic Statistics Report* section for more information.
- **Spare Gates Report**—Refer to the *Spare Gates Report* section for more information.
- **Cross Reference Report**—Refer to the *Cross Reference Report* section for more information.

If you select one of the above reports in a circuit that has multiple variants, the **Variants Filter** dialog box appears.



1. Include or exclude each variant from the report using the **Status** drop-down list.
2. Enable the **Include All Variants** checkbox if you wish to include all variants in the report.

13

Transfer/Communication

This chapter explains how to use Multisim to transfer either circuit schematics or the results of simulation.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

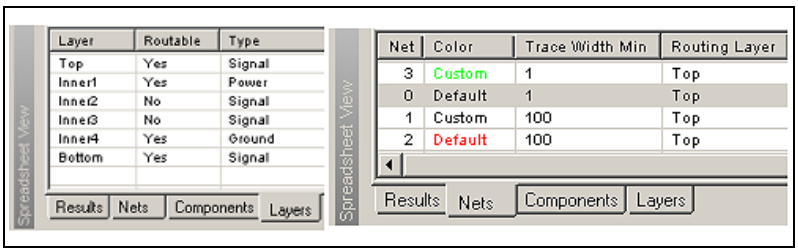
Introduction to Transfer/Communication

Multisim makes it easy to transfer schematic and simulation data to and from other programs for further processing. Multisim can combine schematic information and simulation data for transfer together. For example, when transferring your schematic to perform a PCB layout, Multisim can include optimized trace width information (calculated using the **Trace Width Analysis** during simulation).

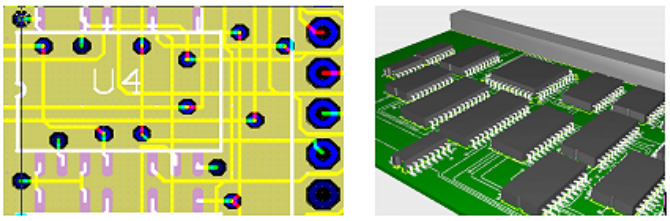
Exporting to PCB layout

Multisim provides features that integrate with many PCB layout programs, and it is particularly well-integrated with its sister product, Ultiboard.

In Multisim you define PCB Layers, and then constrain nets to be routed on these layers. You can also indicate for specific nets the desired, min and max trace widths, min and max trace lengths, minimum distances to other traces, pads, vias, and so on. As well, Multisim will keep track of power and ground nets, and constrain them to be routed only on layers of the appropriate type.



When the schematic has been prepared, the design will be flattened and consolidated, and transferred to Ultiboard for board layout. In Ultiboard you design the board shape and size, place the components on the board, and route the nets as copper traces on as many layers as you like, placing vias to connect traces between layers.



The Multisim and Ultiboard databases also contain full 3D mechanical CAD information for boards and parts. A 3D view of the board enables you to quickly preview the mechanical properties of the board. A small mechanical CAD package in Ultiboard allows for quick casing designs. For larger jobs, the 3D board information can be exported to popular mechanical CAD packages.

In order to enable better routing, Ultiboard may wish to swap pins of components (for example, swap the two inputs of an AND gate), or may wish to swap sections within a component (for example, use a different gate in a QUAD AND chip). Multisim provides Ultiboard with all the information required to do this.

Changes made in Ultiboard (either automatic ones or manual ones) can be “back annotated” to Multisim. For example, if a component is renamed, swapped with one another, pins are swapped, parts are deleted, and so on. The changes will be made into the schematic in Multisim as well. Likewise, subsequent changes to the schematic in Multisim can be “forward annotated” to Ultiboard, in order that progress to date on a board design is not lost if a small change in the schematic is made. Both programs have a feature called “cross probing”, which enables parts or nets in one application to be quickly found in the other.

Transferring from Multisim to Ultiboard for PCB Layout

One of the most common applications to which you may want to transfer data is a PCB layout program. Ultiboard is one of the industry’s leading PCB layout tools and offers many advantages over other layout programs, including trace width optimization synchronized with Multisim simulation.

Caution If you wish to keep analog and digital grounds separate during the PCB layout process, be sure that the **Connect digital ground to analog ground** checkbox in the **PCB** tab of the **Sheet Properties** dialog box is *de-selected* before exporting your schematic to Ultiboard. Refer to the *Sheet Properties - PCB Tab* section of Chapter 1, *User Interface*, for more information.

Complete the following steps to transfer a circuit design from Multisim to Ultiboard 10:

1. Select **Transfer»Transfer to Ultiboard 10**. A standard Windows **Save As** dialog box appears.
2. Specify the name and location of the file to be created and click **Save**. Multisim creates files that can then be loaded into Ultiboard.
3. Load the created files into Ultiboard, following the instructions in the *Ultiboard User Manual*.

Note If you have an earlier version of Ultiboard, select **Transfer»Transfer to Ultiboard 9 or earlier**.

Transfering to Other PCB Layout Packages

If you are using a PCB layout package other than Ultiboard, you can create files in the necessary formats for transfer to the following third party layout packages:

- OrCAD
- PADS Layout 2005
- P-CAD
- Protel

Complete the following steps to transfer the circuit design to a third party layout package:

1. Select **Transfer»Export to PCB Layout**. A standard Windows **Save As** dialog box appears.
2. Navigate to the desired folder, enter a file name, choose the desired manufacturer from the drop-down list and click **Save**. Multisim creates a file of the appropriate format that can then be loaded into the layout package of your choice.

Note Multisection components in different subcircuits, hierarchical blocks or multi-pages of the same design may be placed on the same IC even though they may have different RefDes's (Reference Designators).

Transfering to PADS Layout 2005

All of the components shipped with Multisim map to either a valid PADS footprint, or a generic PADS footprint that will allow you to retain the net connections. However, for this to work properly, two additional PADS files containing these generic footprints must be imported into the PADS library.

Caution You must be careful if you change a component's footprint or create a new one that the component has a valid PADS footprint or generic PADS footprint. Otherwise, the component and the nets attached to it will be deleted by PADS.

Complete the following steps to import the custom PADS library:

1. In PADS Layout, go to **File»Library** to display the **Library Manager**.
2. Click on the **Create New Lib...** button and give the library a name (for example, Multisim Library).
3. Under the **Library** drop-down list make sure the newly created library is selected.
4. Click on the **Decals** button and click the **Import** button.
5. Browse to `...INSTALLDIR\PADS\`, select the `Generic.d` file and click **OK**.
6. Click on the **Parts** button and click the **Import** button.
7. Browse to `...INSTALLDIR\PADS\`, select the `Generic.p` file and click **OK**.

The new parts are now part of the PADS library.

After exporting to PADS Layout, you may map any components with generic PADS footprints to existing PADS footprints. You can also use the **PADS Footprint** column in the **Select a Footprint** dialog box to map generic PADS footprints (or any other type of footprint) to existing PADS footprints before exporting.

Forward Annotation

Forward Annotation lets you annotate changes made to a schematic file in Multisim to its existing Ultiboard file. If you wish to re-number a reference designator, we recommend that you do this from Ultiboard, and back annotate.

Complete the following steps to forward annotate changes to Ultiboard 10:

1. Select **Transfer»Forward Annotate to Ultiboard 10**. A standard **Save As** dialog box appears.
2. Enter the desired filename and filepath and click **Save**. This file can be opened in Ultiboard and used to update the circuit information.

Note If you have an earlier version of Ultiboard, select **Transfer»Forward Annotate to Ultiboard 9 or earlier**.

Back Annotation

Back Annotation allows you to incorporate changes to a circuit that were made in Ultiboard (for example, deleted components) into the corresponding Multisim circuit file.

Complete the following steps to back annotate changes made in Ultiboard:

1. Select **Transfer»Back Annotate from Ultiboard**. A standard **Open** dialog box appears.
2. Select the desired `.log` file and click **Open**. The **Annotation Log** dialog box appears.
3. Select the changes that you wish to back annote to your Multisim file and click **OK**.

Note The circuit file must be open before you use this command. If a changed part is in a hierarchical block or a subcircuit with multiple instances, changes will be ignored.

Exporting Simulation Results

You can export simulation results from the **Grapher** to MatchCAD, Excel or LabVIEW. You must have the required software installed on your computer if you are exporting to either MathCAD or Excel.

Exporting to MathCAD

You can export the results of your simulation to MathCAD, allowing you to perform sophisticated mathematical operations on your data.

Note This function is only available if you have MathCAD installed on your computer.

Complete the following steps to export the simulation results to a MathCAD session:

1. Select **View»Grapher**. The **Grapher** appears, showing the results of your simulation and/or analysis.
2. Click the **Transfer to MathCAD** button. The **Select Traces** dialog box appears. The contents of this dialog box change depending on the traces that are found in the tab you have selected in the **Grapher**.
3. Use the checkboxes to select the traces you wish to export to MathCAD and click **OK**.

Note If you place the cursors on the graph (via the **Show/Hide Cursors** button), the output data will be limited to the data that is located between the two cursors.

4. A new MathCAD session is started.

Note MathCAD will shut down when Multisim shuts down.

Exporting to Excel

You can export your simulation results to Excel, allowing you to use the data for further processing in a spreadsheet.

Note This function is only available if you have Excel installed on your computer.

Complete the following steps to export the simulation results to an Excel spreadsheet:

1. Select **View»Grapher**. The **Grapher** appears, showing the results of your simulation and/or analysis.
2. Click the **Transfer to Excel** button. The **Select Traces** dialog box appears. The contents of this dialog box change depending on the traces that are found in the tab you have selected in the **Grapher**.
3. Use the checkboxes to select the traces you wish to export to Excel and click **OK**.

Note If you place the cursors on the graph (via the **Show/Hide Cursors** button), the output data will be limited to the data that is located between the two cursors.

4. A new Excel spreadsheet is created, with data from the x coordinates in column one and data from the y coordinates in column two.
5. If desired, save the Excel spreadsheet.

Saving to a Measurement File

You can save the results of your simulation to either a text-based (`.lvm`) or binary (`.tdm`) measurement file, and use applications like National Instruments LabVIEW and DIAdem to compare simulated output with actual circuit output.

Complete the following steps to save the simulation results as a measurement file:

1. Select **View»Grapher**. The **Grapher** appears, showing the results of your simulation and/or analysis.
2. Click the **Save to Measurement File** button. A file browser appears.

3. Select the desired file type from the drop-down list. Available file types are:
   - Text-based measurement files (`*.lvm`)—Files such as those created in NI LabVIEW.
   - Binary measurement files (`*.tdm`)—Files used to exchange data between National Instruments software, such as LabVIEW and DIAdem. Note that when you save data as this file type, two files are created: a header file (`.tdm`) and a binary file (`.tdx`).
4. Select the desired filepath, enter a filename and click **Save**. The **Data resampling settings** dialog box appears.
5. Change the following settings as desired:
   - **Do not resample** checkbox—Appears for `.tdm` files only. Enable if you do not wish to resample the data. The other options are disabled.
   - **Resample data** checkbox—Appears for `.tdm` files only.
   - **Interpolation mode**—Select one of: **Coerce**; **Linear Interpolation**; **Spline Interpolation**. Refer to the *Interpolation Modes* section of Chapter 10, *Analyses*, for information about each.
   - **Δx (in seconds if time-domain data)**—The sampling period to use for resampling.
   - **1/Δx (in Hz if time-domain data)**—The sampling rate to use for resampling.

Note You can change either **Δx** or **1/Δx**. The other will change accordingly.

   - **Estimated file size**—This read-only field changes as you change either **Δx** or **1/Δx**.

6. Click **OK** to close the dialog and save the file.

Exporting a Netlist

You can export a netlist for your circuit.

Complete the following steps to export a netlist of your design:

1. Select **Transfer»Export Netlist**. A standard Windows **Save As** dialog box appears.
2. Select the filepath and filetype, enter the filename and click **Save**. The netlist is saved.

Importing Files with Other Formats

Files with the following formats can be opened with Multisim:

- Multisim 10 Files—`*.ms10`. Opens Multisim 10.0, Multisim 10.0.*x*, and Multisim 10.1 files.
- Older Multisim Files—`*.ms9`, `*.ms8`, `*.ms7`, `*.msm`
- Electronics Workbench 5 Files—`*.ewb`
- Multisim 10 Project Files—`*.mp10`
- Older Project Files—`*.mp9`, `*.mp8`, `*.mp7`
- EWB Database Update Files—`*.prz`
- SPICE Netlist Files—`*.cir`
- Orcad Files—`*.dsn`
- Ulticap Files—`*.utsch`
- EDA Parts Update Files—`*.edp`, `*.mxm`, `*.msml`

Note When opening files from earlier versions of Electronics Workbench or Multisim, Reference Designators may be renamed to ensure that all instances are unique.

Complete the following steps to open a file of any of the above-listed types except Electronics Workbench 5 or Ulticap files:

1. Select **File»Open**. A standard **Open** dialog box displays.
2. Select the desired file type from the **Files of Type** drop-down list.
3. Highlight the desired file and click **Open**. The file is opened in Multisim.

Opening an Electronics Workbench 5 File

Complete the following steps to open a `.ewb` file:

1. Open the file in the usual manner. The **File Loading Options** dialog box displays, with the default **Horizontal Scaling** and **Vertical Scaling** settings of 120, that is 120% of the original spacing.

   This indicates that there will be 20% more space between the components than in the original `.ewb` file. This is because the components in Multisim are larger than in Electronics Workbench 5.
2. If desired, increase or decrease the spacing between the components by increasing or decreasing the **Horizontal Scaling** and **Vertical Scaling** values.
3. Click **OK** to import the file. A dialog appears asking you to wait while the circuit is imported.

Opening an Ulticap File

Complete the following steps to open an Ulticap schematic file:

1. Open the file in the usual manner. The **Ulticap Import** dialog displays.
2. Select the desired options in the **Save to Database Options** box:
   - **Do not save imported parts**—Imports the Ulticap parts without saving them to any of the Multisim databases.
   - **Save imported parts**—Saves the imported Ulticap parts to the selected Multisim database.
3. In the **Use Grid Size** box, select one of:
   - **50 Mil**—Places the Ulticap parts and the connected wires on a 50 mil grid. This will result in fewer importation errors, but some parts may appear quite large.
   - **100 Mil**—Places the Ulticap parts and the connected wires on a 100 mil grid. Symbols will be smaller in size, but more importation errors may occur after the file is imported.
4. Click **OK** to import the file. A status message displays during importation of the Ulticap file. If desired, click **Abort** to cancel.

Importing a SPICE or Cadence® PSpice® Netlist

In addition to the **File»Open** command, there is a second method for importing SPICE or Cadence® PSpice® netlists. The `source` command, under the **Simulate»XSPICE Command Line Interface** menu, is available for advanced users who need the flexibility offered from the command line. With this method, models and commands in the netlist file are interpreted, without substitution, by the simulation engine as they are represented in the netlist.

Complete the following steps to import a SPICE or Cadence® PSpice® netlist from the command line:

1. Select **Simulate»XSPICE Command Line Interface** to display the XSPICE Command Line dialog.
2. Type the `source` command and the filename (for example, `source rc.cir`) in the bottom line and press ENTER. This will load the netlist along with the models and connectivity information into memory. You will not see the circuit on the workspace but you can continue to execute other commands, such as `tran`, and display the results on the grapher with the `plot` command.

14

RF

This chapter contains descriptions of the components, model makers, instruments and analyses of Multisim's RF design functionality, as well as a tutorial demonstrating their use and some specific examples of RF functionality.

Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition.

Introduction to the Multisim RF Module

Multisim RF provides fundamental RF circuit design features needed to design, analyze and simulate RF circuits.

It consists of:

- RF-specific components, including customized RF SPICE models
- model makers for creating your own RF models
- two RF-specific instruments (**Spectrum Analyzer** and **Network Analyzer**)
- several RF-specific analyses (circuit characterizer, matching network cells, noise figure).

Elements of the RF design functionality are fully integrated into Multisim. The instruments, analyses and components are installed in the same places and invoked in the same way as all other instruments, analyses and components. The RF components have their own group which is accessed via the **RF** button on the **Components** toolbar and the RF instruments are accessed through the **Instruments** toolbar.

RF Components

Electronics components fall into two categories—lumped components and distributed components. Lumped components are smaller than the wavelength, where:

$$\lambda = \frac{c}{f}$$

In these cases, the wavelength of the voltages and currents at which they are operating is significantly larger than the components themselves. Ohm's Law, in this case, is valid. For example, a 1/4 watt resistor is about 0.270 inches long and 0.090 inches in diameter.

RF components, on the other hand, are most often distributed elements, where the phase of a voltage or current changes significantly over the physical extent of the device, because the device dimensions are similar to, in some cases even larger than, the wavelength. Standard circuit theory is therefore not always applicable to circuits that are working at a few MHz to above 1 GHz.

The usual models for lumped components are often not valid for RF devices. For example, a capacitor can behave like an inductor or an inductor can behave like a capacitor at high frequencies.

RF components exhibit parasitic effects, and have models different from those used in low frequencies. Connections between two nodes at high frequencies exhibit different behavior from those at low frequencies, and are modeled using capacitances and inductances. These connections behave as transmission lines when implemented on a printed circuit board (PCB). The board itself becomes part of the circuit, interfering with the normal circuit function. Consequently, low frequency circuit simulation EDA tools can become unreliable at higher frequencies.

Standard RF components include capacitors, inductors, toroids, ferrite beads, couplers, circulators, transmission lines or striplines, waveguides, and high frequency active devices such as transistors and diodes. More complex components, such as quadrature hybrids, mixers, filters, and attenuators, are built using these standard components. This chapter deals with the standard components and their models in high frequencies.

Multisim's RF Components

The RF group contains parts and models specifically built for accuracy at higher frequencies. This ability to handle higher frequencies helps overcome a typical problem with SPICE models, which tend to perform poorly at such frequencies.

To access these parts, click the **RF** button in the **Components** toolbar.

There are several component families in the RF parts group, including: RF capacitors, RF inductors, RF NPN BJTs, RF PNP BJTs, RF MOSFETs, RF tunnel diodes, RF striplines/waveguides and ferrite beads.

Families containing components with wide commercial availability (for example, **RF_BJT_NPN**) have a large number of components within them. Families containing components that are not readily available "off-the-shelf" (for example, RF inductors) contain only a few components. The latter were modeled using Multisim's RF **Model Makers**, which you can also use to customize your own parts. Refer to the *RF Model Makers* section for more information.

Theoretical Explanation of the RF Models

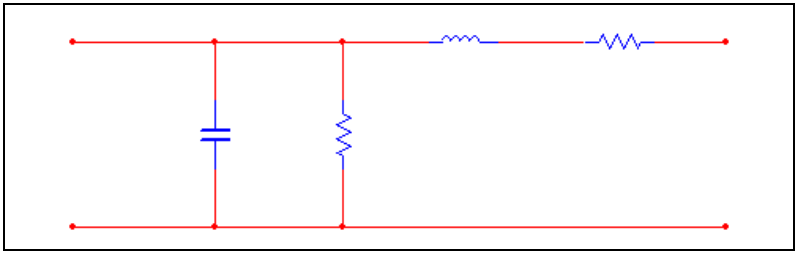
This section explains some of the scientific theory behind the operation of RF components and why they must be modeled differently at higher frequencies. You do not need to understand this information to use the RF components in Multisim, but it may be of interest to you.

Striplines/Microstrips/Waveguides

Wires are used to connect two internal nodes on a circuit and show no difference in phase and magnitude between two connecting nodes. Wires, however, behave differently at RF frequencies depending on their length and diameter. One of these behaviors is called the "skin effect", explained below.

A conductor, at low frequencies, uses its entire cross-sectional area as a transport medium for charge carriers. As the frequency is increased, an increased magnetic field at the center of the conductor presents an impedance to the charge carriers, thus decreasing the current density at the center of the conductor and increasing it around its perimeter. This effect is called the "skin effect", and occurs in all conductors, including resistor leads, capacitor leads, and inductor leads. As the frequency increases, this effect is more pronounced.

A simple wire connecting two nodes in high frequencies behaves as a transmission line. The following figure shows the equivalent circuit of a transmission line. There are four components. The capacitor is the result of an actual capacitance existing between the center of the conductor and the ground. Between these two plates is the dielectric, which is not perfect. This leakage is modeled using conductance G and is given per unit length of line. Also, due to the resistance of the conductor itself, there is a series resistance R. Its value depends on the resistivity of the material used, the length, the cross-section of the conductor, and the skin effects.



Every transmission line has a resistance, called its “characteristic impedance”. Most microwave systems have a characteristic impedance of 50 ohms. This value is a compromise between maximum power handling capability and minimum attenuation. At 50 ohms, there is a reasonably low attenuation, and adequate power handling capability.

If the outer diameter of the conductor of a coaxial line is shown by “D”, and the inner diameter is shown by “d”, and ε is the dielectric constant of the cable, the characteristic impedance is calculated by the following formula:

$$Z_0 = \frac{138}{\sqrt{\varepsilon}} \log_{10}\left(\frac{D}{d}\right)$$

The components C and L in the figure above are calculated as follows:

$$C = \frac{7.354\,\varepsilon}{\log_{10}(D/d)} \qquad ((PF)/(ft))$$

$$L = 0 - 1404 \log_{10}(D/d) \quad ((\mu H)/(ft))$$

A stripline is a useful form of a transmission line. The stripline consists of a conducting strip lying between, and parallel to, two wide conducting planes. The region between the strip and the planes is filled with a uniform dielectric.

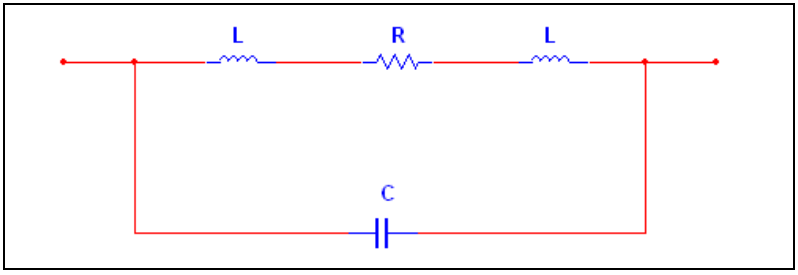
Microstrips are easy to fabricate using photolithographic processes. At the same time that a transistor is placed on top of the board, for example, a microstrip can also be placed. Microstrips are, therefore, easily integrated with other passive and active devices. A conductor of width W is printed on a thin, grounded dielectric substrate of thickness “d” and relative permittivity “u_r”.

A waveguide is a structure, or part of a structure, that causes a wave to propagate in a chosen direction.

If the waveguide boundaries change direction, the wave is constrained to follow. Waveguides come in a variety of types: simple parallel plate structure, cylindrical structures with conducting boundaries, rectangular waveguides, and circular waveguides. A transmission line or a stripline is a special case of waveguide.

RF Resistors

Resistors find many applications as terminators or attenuators. The equivalent circuit of a resistor at radio frequency is shown in the following figure. The inductors are calculated using the physical geometry of the resistor.



$$L = 0.0021\left[2.3\log\left(\frac{4l}{d} - 0.75\right)\right](\mu H)$$

l = length of wire in cm

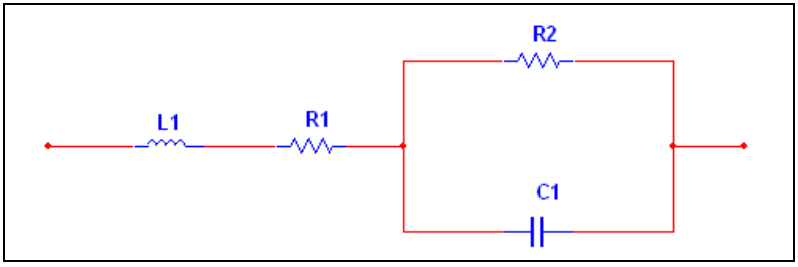
d = diameter in cm

RF Capacitors

Capacitors are used for interstage coupling, for bypassing, in resonant circuits, and in filters. RF capacitors must be chosen carefully to ensure the best performance for specific applications. RF capacitors consist of two metal plates separated by a dielectric. The capacitance of an *ideal* capacitor has a direct relationship with the area (A), and is inversely proportional to the thickness of the dielectric (d). Its relationship is expressed in the following formula, where ε is the dielectric constant of the dielectric material.

$$C = (d\varepsilon)\frac{A}{d}$$

The actual capacitor shows imperfection. One type of capacitor is modeled as shown in the following figure.



To find the numerical values of the ideal elements in the model above, a number of factors are considered.

Let ϕ represent the phase of current compared to the voltage. This phase is ideally 90, but is smaller for real components. The power factor (PF) is defined as COS (ϕ). This factor is a function of temperature, frequency, and the dielectric material. The power factor is usually used to describe the capacitor in low frequencies.

This factor in higher frequencies is sometimes referred to as the dissipation factor. This factor describes how much power is dissipated, lost, or transformed to heat energy in RF frequencies. Another factor that defines the quality of the capacitor is closely related to power factor and is called Q. This factor is the reciprocal of dissipation factor. The larger the Q, the better the capacitor.

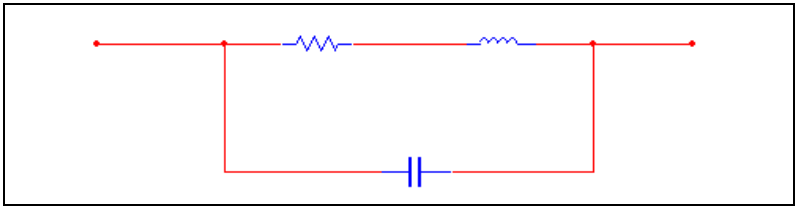
No dielectric material is perfect. Therefore, there is always leakage current between two plates. This behavior is best described by Rp which is usually around 100,000 MOhm. The series resistor is the AC resistance of the

capacitor at high frequencies, and is obtained using COS (ϕ)/Cω * 1e6. Here, ω =2*pi*f.

There is a frequency point above which the capacitor starts to behave like an inductor.

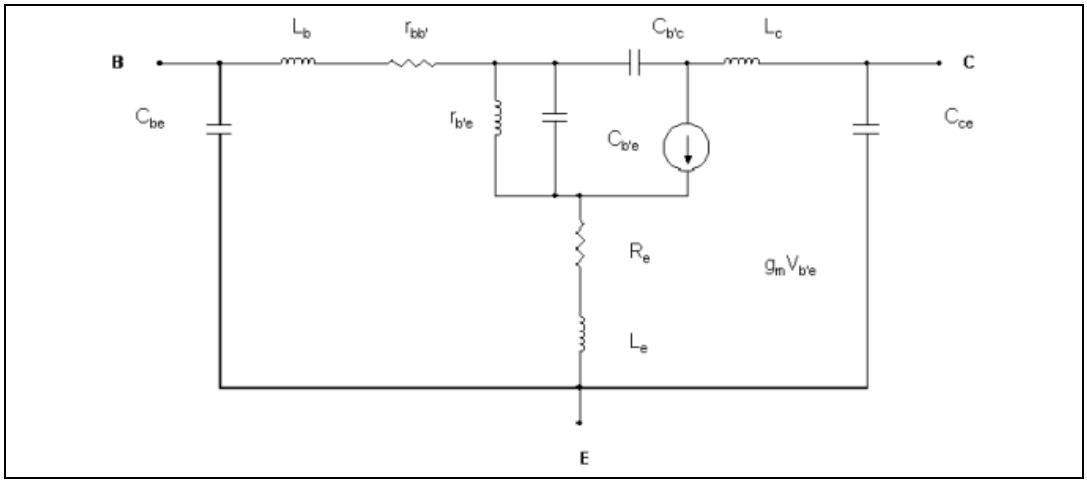
RF Inductors

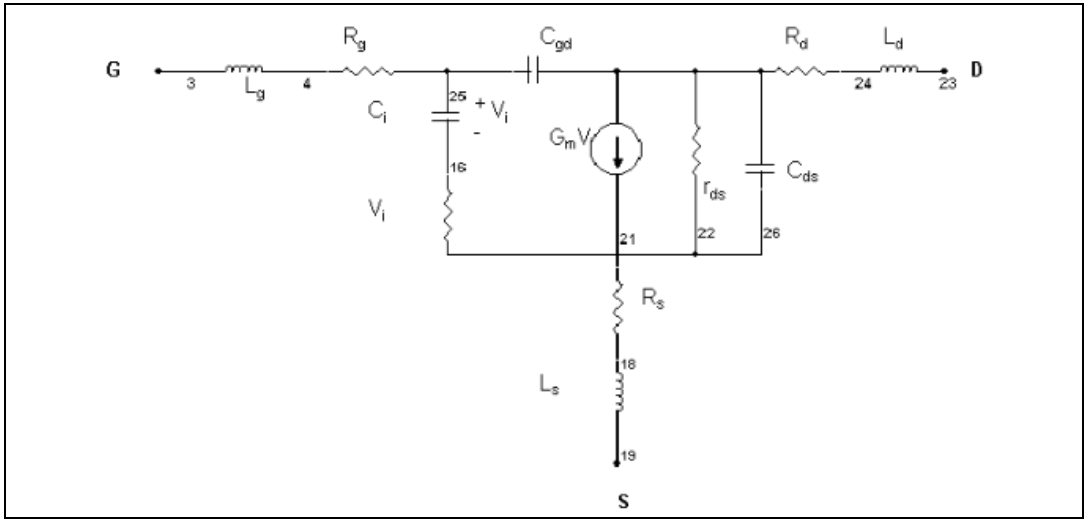
Inductors are extensively used in resonant circuits, filters, and matching networks. The following figure shows a typical inductor modeled for RF frequencies. An inductor is a wound or coiled wire. Each two windings are at close proximity, which creates a distributed capacitor, Cd. The inductor behaves like a capacitor at high frequencies. There is always a series resistance which prevents the coil from resonating. The ratio of an inductor's reactance to its series resistance is often used as a measure of the quality of the inductor. The larger the ratio, the better the inductor.



Devices

In low frequencies, active devices are modeled using a number of ideal components such as resistors and capacitors. In high frequencies, each of these ideal components should be replaced by its equivalent, as discussed earlier. For example, a resistor should be replaced by a resistor in series with an inductor. Some simplifications reduce redundant components. Two inductors in series, for example, can be replaced by one inductor. A typical equivalent circuit of an RF transistor is shown in the following figures.





The cut-off frequency f_c can be derived from the equivalent circuit and is inversely proportional to the transit time τ_c:

$$f_c = \frac{g_m}{2\pi C_i} = \frac{1}{2\pi \tau_c} = \frac{v_s}{2\pi L_g}$$

where L_g is the effective length of the gate, and v_s is the saturation velocity at which electrons travel.

RF Instruments

Multisim has two key instruments for successful RF circuit design and analysis: the **Spectrum Analyzer** and the **Network Analyzer**.

Spectrum Analyzer

The **Spectrum Analyzer** is capable of measuring a signal's power and frequency components, and helps determine the existence of harmonics in the signal.

One area that has an interest in spectrum measurement is communications. For example, cellular radio systems must be checked for harmonics of the carrier signal that might interfere with other RF systems.

Usually the frame of reference in signal analysis is time. In that case, an oscilloscope is used to show the instantaneous value as a function of time. Sometimes a sine waveform is expected but the signal, rather than being purely sinusoidal, has a harmonic on it. As a result, it is not possible to measure the waveform's level.

However, the **Spectrum Analyzer** displays its measurements in the frequency domain rather than the time domain. If the same signal (with harmonics) is displayed on a **Spectrum Analyzer**, the amplititude of its fundamental frequency displays, as well as its frequency components (harmonics).

Time domain measurements such as rise and fall times, pulse width, repetition rates, and delays, cannot be easily obtained in frequency domain measurements. Therefore, both techniques are important.

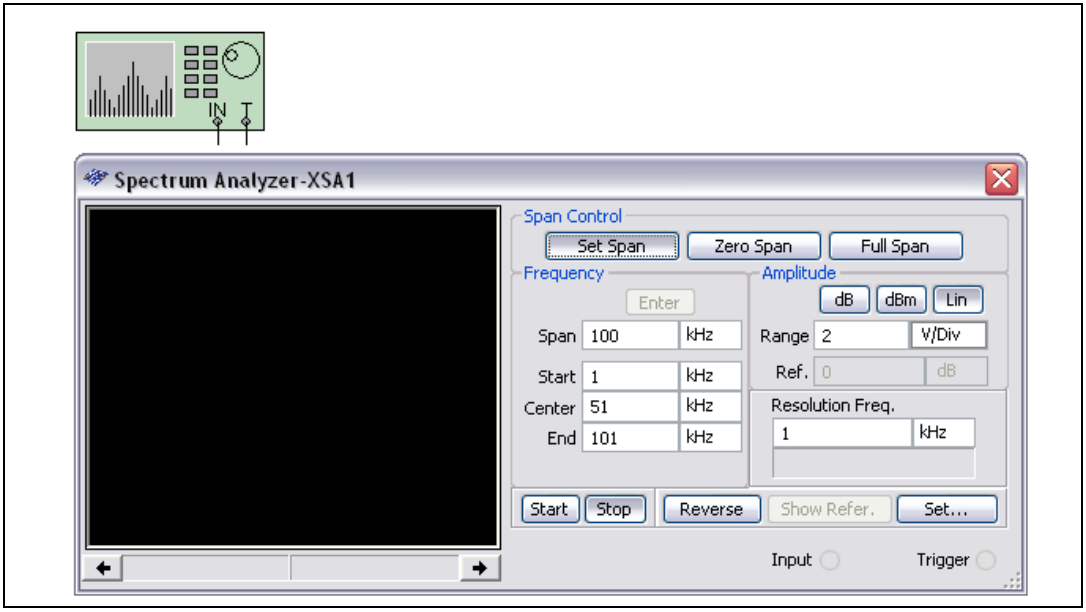
Using Multisim's Spectrum Analyzer

Multisim's **Spectrum Analyzer** does not generate the noise normally expected in a real-world spectrum analyzer. In reality, the noise generated by a spectrum analyzer (due to the random electron motion through the various components of an analyzer) is amplified by the various gain stages in the analyzer, and ultimately appears on the display as a noise signal below which measurement cannot be made. With Multisim's **Spectrum Analyzer**, no additional noise is introduced by the instrument itself.

A number of parameters characterize a **Spectrum Analyzer**:

- frequency range in which the instrument operates
- frequency spans
- reference level
- measurement range.

These are all represented on the Multisim **Spectrum Analyzer**, and must be set manually.



Frequency Range

Frequency range is the range of frequencies over which the **Spectrum Analyzer** will analyze signals. There are two frequencies that you manually set: f-start (minimum value 1kHz) is entered in the **Start** field and f-end (maximum value of 4 GHz) is entered in the **End** field. Zero frequency is not allowed for any of the frequency settings.

Frequency Spans

This parameter indicates the frequency range to be displayed:

- **Set Span**—The frequency span is determined using either span control or frequency control. Refer to the *Frequency Analysis* section for more information.
- **Zero Span**—A single frequency defined by the **Center** field is displayed.
- **Full Span**—The entire instrument range of 1 kHz to 4 GHz is displayed.

Frequency Analysis

There are two methods to select the frequency range:

- span control
- frequency control.

Span Control

This technique sets the span and the center frequencies, that is, f-span and f-center.

To set the span and the center frequencies, click **Enter** (in the **Frequency** area of the instrument's display) to automatically calculate the values of f-start and f-end using the following expressions:

f-start = (f-center – f-span / 2)

f-end = (f-center + f-span/2)

Frequency Control

You can define the starting and ending frequencies manually. In order to do this, you need to enter the numerical values of frequencies in the **Frequency** area of the dialog box. Their values should be non-zero values. When **Enter** is clicked, the center frequency (f-center) and the range of frequency displayed on the **Spectrum Analyzer** (f-span) are calculated automatically. The relationship among these parameters is expressed as follows:

f-center = (f-start + f-end)/2

f-span = (f-end - f-start)

These two techniques are interrelated, that is, it is not possible to set all four parameters independently. Both techniques are useful. For example, if you want to see frequency components around one specific frequency such as 100 Mhz +/– 100 kHz, then the frequency control technique is easier to apply. The center frequency, in this example, is 100 Mhz, and the span is 2*(100 kHz) = 200 kHz.

Amplitude Range

You can set the amplitude range of the signal visible on the screen by choosing one of the following three options in the **Amplitude** area:

- **dB**—Stands for 20*log10(V), where log10 is the logarithm in base 10, and V is the amplitude of the signal. When this option is used, the signal is displayed by “dB per division”, shown in the right-hand side of the **Spectrum Analyzer**. The dB reading is of interest when measuring the power of the signal.
- **dBm**—Stands for 10*log10(V/0.775). Zero dBm is the power dissipated in a 600 Ohm resistor when the voltage across it is 0.775 V. This power is equal to 1 mW. If the level of a signal is +10 dBm, it means that its power is 10 mW. When this option is used, the signal power is displayed based on the reference of 0 dBm. For applications in which the terminating resistor is 600 Ohm, such as in telephone lines, it is more convenient to read dBm as it is directly proportional to the power dissipation. However, in dB, you need to include the value of the resistor to find the dissipated power in the resistor. In dBm, the value of the resistor has been accounted for already.
- **Lin**—Selects a linear display of the signal. To change the maximum amplitude displayed on the screen, enter a voltage value in the **Range** field.

Reference Level

The reference level is used to set the range of the input signal that can be displayed on the screen.

The axes of the **Spectrum Analyzer** are not marked by units and values. You can read the frequency and the amplitude of each point displayed on the screen by using the cursor. You can also right-click on the cursor to display a pop-up menu that lets you move the cursor to a precise point on the graph. When the cursor is moved and placed on the point of interest, the frequency and the amplitude in V, dB, or dBm are displayed below the graph.

You can observe more than one frequency and evaluate the results for the entire frequency range shown. You may want to know when the amplitude (in dB or dBm) of some components is above a certain limit in dB or dBm. For example, assume you are interested in the (–3dB) amplitude. By locating (–3dB) points you can estimate the bandwidth of the amplifier. By clicking **Show Refer.**, you can set the reference level to (–3dB) and, using the cursor at the same time, you can find the lower edge and upper edge of the pass band.

You can also find out whether the amplitude of the signal is less than a certain value for a certain band of frequency. To do this, observe signals on the **Spectrum Analyzer**, and use the reference button. The maximum reference value in dB is set to (+30 dB). **Show Refer.** is available only if either **dB** or **dBm** are activated.

Frequency Resolution

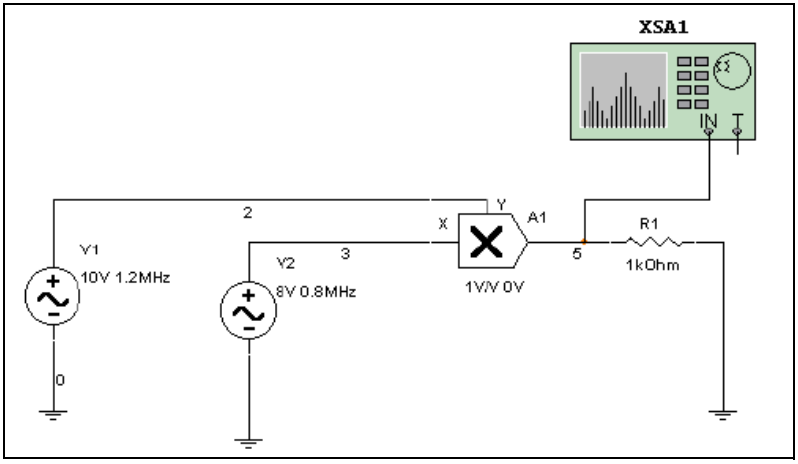
The **Resolution Freq.** is initially set to a minimum value of $\Delta f = f\_end/1024$. However, you can change it to a greater value and observe the spectrum. You must select the frequency resolution so that the frequencies are integer multiples of frequency resolutions.

Note For an accurate reading, the frequency components should not be below Δf.

Examples

Example 1

The following figure shows a mixer, which is often used in communications applications.



There are two input sinusoidal waveforms. Their frequencies are 0.8 MHz and 1.2 MHz. The amplitudes are set at 8 V and 10 V, respectively. Note that the amplitude is the peak value of the sinusoidal waveform—it is not the RMS value of the waveform. The mixer is set to multiply the signals with unity gain without introducing an offset in either of the input signals. You can expect to find two components at the output placed at (1.2 + 0.8) = 2 MHz, and (1.2 – 0.8) = 0.4 MHz.

Complete the following steps to try this example for yourself:

1. Construct the network as shown above.
2. Double-click on the multiplier and set the gains to 1 and the offsets to zero.
3. Double-click on the **Spectrum Analyzer** and initialize it using one start and end frequency (*Example 3* shows another way to initialize the **Spectrum Analyzer**).
   - Set **Span** to 3MHz and **Center** to 1.8 MHz.
   - Click **Enter**. The frequency value of f-start is automatically set to (1.8 MHz – 3/2 MHz) = 300 kHz. The frequency value of f-end is automatically set to (1.8 MHz + 3/2 MHz) = 3.3 MHz.

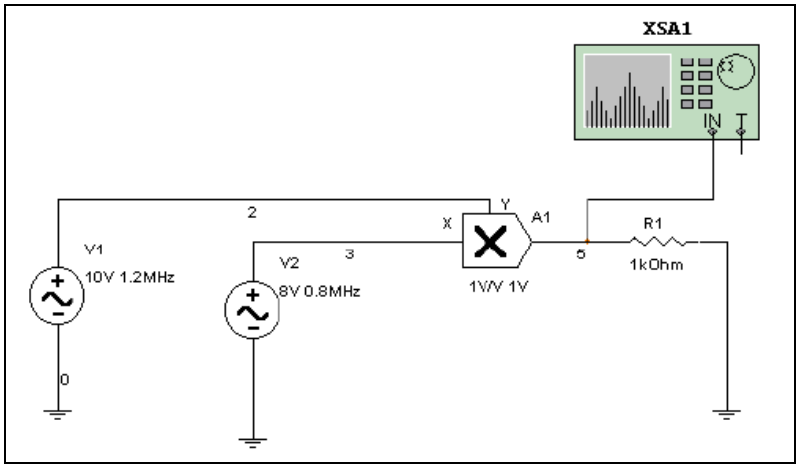
- Since the amplitude of the component is around (8*10)/2 = 40 V, set the amplitude **Range** to 100 V in **Lin** mode.

4. Click **Start** and wait until the signal stabilizes.

The **Spectrum Analyzer** starts performing the Fourier Transform of the input signal in time domain. However, since it begins with only a few samples, it does not provide accurate results initially. You must wait until the screen is refreshed a few times to obtain accurate readings of the frequency components and their magnitudes. At this time, the internal frequency resolution is equal to the user-defined frequency resolution. Both of these values are shown on the device. Using the cursor on the screen, you can read the amplitude and frequency of each component. In the example, the readings are the same as the calculated values, that is, two frequency components at 2 MHz and 0.4 MHz, with 40 V magnitude.

Example 2

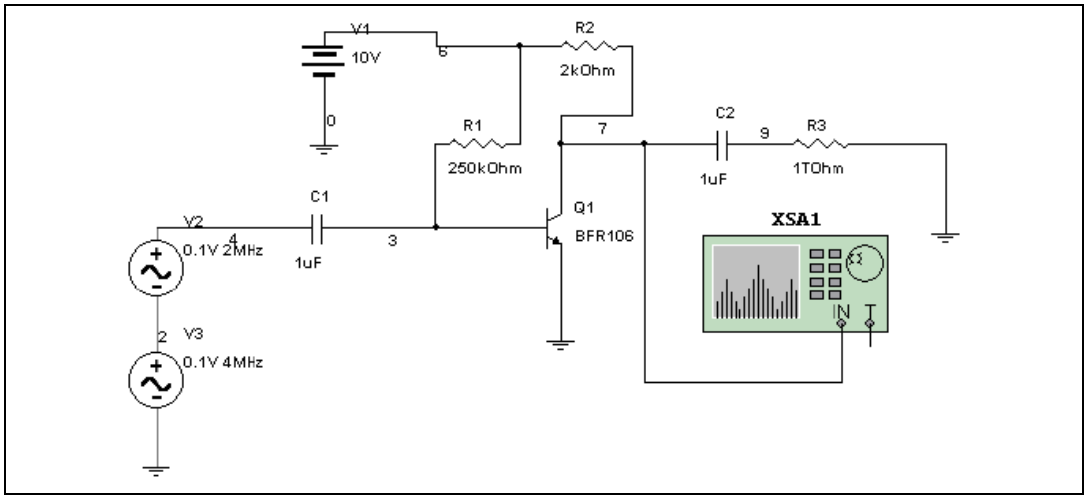
In order to run another example, you must stop the simulation running for the previous example. The second example has the same circuit structure, as shown below.



However, DC offsets (1 V) to the input and the output signals are introduced. Due to the offsets, there are components at DC, as well as 0.8 MHz and 1.2 MHz. The **Spectrum Analyzer** shows the components and their amplitude on the screen if the same setting used in the previous example is also used here.

Example 3

This example considers an active device that works in saturation. This structure is shown below:



Complete the following steps to initialize the **Spectrum Analyzer**:

1. Double-click on the **Spectrum Analyzer**.
2. Set **Start** to 1 kHz and **End** to 11MHz.
3. Click **Enter**. The frequency value of f-span is (11 MHz – 1 kHz) = 10.999 kHz. The frequency value of f-center is (11 MHz + 1 kHz)/2 = 5.5005 MHz.
4. Set **Range** to 2 dB/division, and set **Ref.** to 4 dB to demonstrate the application of reference dB level.

The frequency values of f-start and f-end are set so that the frequency components of interest are captured and their magnitudes can be studied. Using the **Spectrum Analyzer**, you can verify that there are more than two frequency components present at the output node. There are three components above 4 dB—zero frequency, 2 MHz, and 4 MHz. Other frequency components are at higher frequencies and have dB level less than 4 dB.

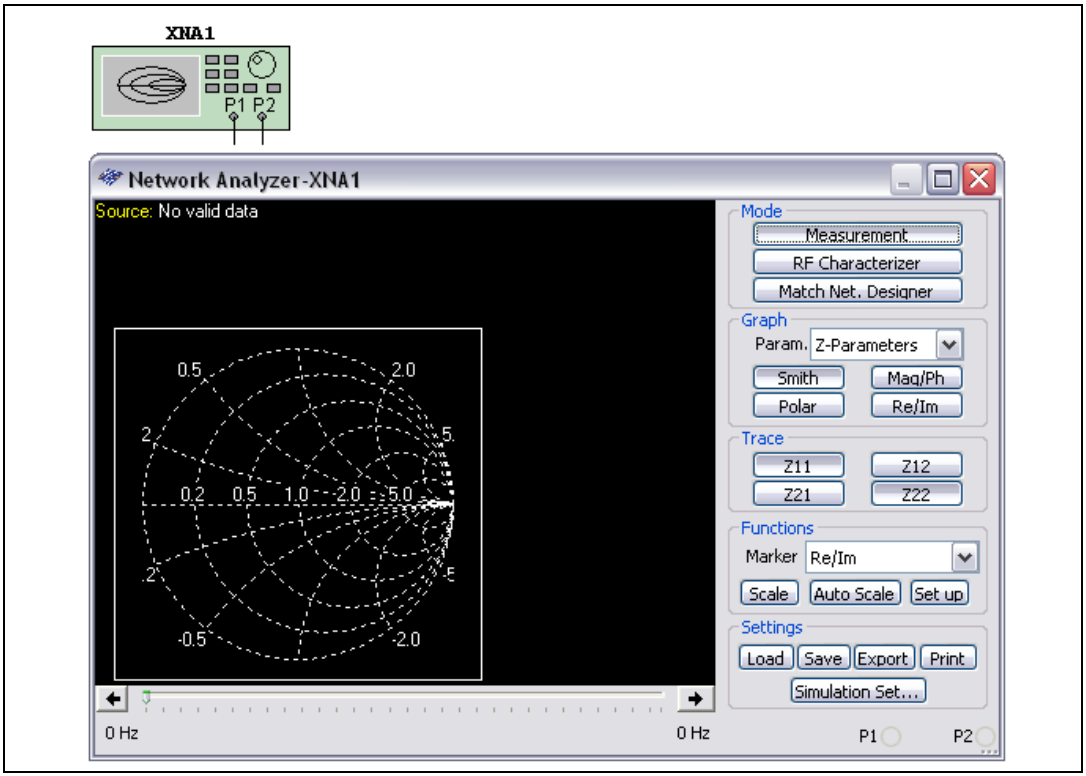
Network Analyzer

The **Network Analyzer** is used to measure the scattering parameters (or S-parameters) of a circuit, commonly used to characterize a circuit intended to operate at higher frequencies. These S-parameters are used for a variety of purposes, including in Multisim RF analyses. The **Network Analyzer** also calculates H, Y, and Z parameters.

Using the Network Analyzer

The circuit is idealized as a two-port network. To properly use the **Network Analyzer**, the circuit must be left open at its input and output ports. During simulation the **Network Analyzer** completes the circuit being analyzed by inserting its subcircuits. You must remove these subcircuits from the circuit before performing other analysis and simulation.

When you start simulation, the **Network Analyzer** automatically executes two AC analyses. The first AC analysis is applied at the input port to compute the forward parameters S11 and S21. The second analysis is applied at the output port to compute the reverse parameters S22 and S12. After the S-parameters are determined, you can use the **Network Analyzer** to view the data in many ways and perform further analyses based on the data.



The **Network Analyzer** is divided into two regions:

- The left region contains a chart display area and a text display area that shows marker and circuit information.
- The right region contains the controls.

Marker Controls

Select how you want data represented from the **Marker** drop-down list:

- Real/Imaginary **(Re/Im)**
- Magnitude/Phase **(Mag/Ph(Deg))**
- dB Magnitude/Phase **(dB Mag/Ph (Deg))**.

Use the left and right arrows below the graph to step through each data point acquired for the circuit. Initially, the frequency points are selected in decade. The default frequency sweep range is between 1 MHz and 10 GHz. You can also use the <Page Up> and <Page Down> keyboard keys to navigate the data set.

Trace Controls

Use the **Trace** buttons to show and hide each individual trace on the chart. When the **Network Analyzer** is opened, S11 and S22 are shown in the trace area. The trace buttons change depending on what parameter or analysis is being looked at. The possible sets in **Measurement** mode are {S11, S12, S21, S22}, {Z11, Z12, Z21, Z22}, {H11, H12, H21, H22}, {Y11, Y12, Y21, Y22}, {K, |Δ|}. The possible sets in **RF Characterizer** mode are {P.G., T.P.G., A.P.G.}, {V.G.}, and {Zin, Zout}. Refer to the *Mode Controls* section for more information about these modes.

Format Controls

Parameter Options

Select the set of parameters to be displayed on the chart. The parameters available depend on the **Mode** setting. In **Measurement** mode, you can choose from S-, Y-, H-, Z- parameters, and stability factors K and |Δ|. In **RF Characterizer** mode, select from Power gains, Voltage gains, and Impedance.

Format Buttons

Use these buttons to display data using different chart formats. The formats available depend on the selected parameter group.

Use the **Scale** button to change the scaling of the current chart. Only Polar plot, Real/Imaginary plot and Magnitude/Phase plot can be changed.

Use the **Auto Scale** button to automatically scale the data so that it can be displayed within the current chart.

Use the **Set up** button to change the various display properties of the **Network Analyzer**.

Data Controls

Use the **Save** button to save the current S-parameter data set to file.

Use the **Load** button to load a previously saved S-parameter data set into the **Network Analyzer**. Once the data is loaded, you can use all the functions provided by the **Network Analyzer** to view and analyze the data. The saved S-parameter file has the file extension `.sp`.

Use the **Export** button to export the data set of the selected parameter group to a text file. For example, if the selected parameter group is Z-parameters displayed in magnitude/phase chart format, Z-parameters will be exported, and their values will be in magnitude/phase format.

Use the **Print** button to print the selected chart.

Mode Controls

From the **Mode** box, click the button for the desired **Network Analyzer** mode:

- **Measurement**—Provides the parameters in different formats.
- **RF Characterizer**—Provides the power gains, voltage gains, and impedances seen from input and output ports.
- **Match Net. Designer**—Opens a new dialog box which is described in the *Matching Network Analysis* section.

Use the first of the above choices to use the **Network Analyzer** to perform measurement (its normal application). Use the second and third options to access the two RF analyses explained in the *RF Analyses* section.

Use the **Simulation Set** button to enter the measurement settings for computing the circuit's S-parameters.

RF Analyses

Multisim includes the following RF analyses:

- **RF Characterizer Analysis**—Helps designers study RF circuits in terms of their power gains, voltage gain, and input/output impedances.
- **Matching Network Analysis**—Provides three options for analyzing circuit behavior.
- **Noise Figure Analysis**—Use to specify how noisy a device is.

RF Characterizer Analysis

Multisim's RF characterizer analysis tool helps designers study RF circuits in terms of their power gains, voltage gain, and input/output impedances. A typical application is an RF amplifier. The source signal at the input of an amplifier is usually provided by a receiver and its power is relatively small. The RF designer often intends to magnify the input signal and provide an output signal in terms of both voltage and current: that is, the output power delivered to the load is considerably higher than that of the

input signal. That is why the power transferability of the designed circuit is of interest. The power gains in Multisim are calculated by assuming that source and load impedances are 50 Ohm. You can change these values by clicking the **RF Param. Set** button to specify that the RF simulator assumes:

Zl = Zo and Zs = Zo or Γs = Γl = 0.

Another aspect of a circuit is the input and output impedances of the amplifier. An RF amplifier usually has more than one stage of amplification. Each stage of the amplifier is loaded by the input port of the next stage.

The loading effect is best understood by studying the input/output impedances. Most engineers would like to design an amplifier which has maximum input impedance in low RF frequencies, to reduce its loading effect on the previous stage. On the other hand, the smaller the output impedance is, the better the output signal would be delivered. In higher RF frequencies, it is desirable to have an output impedance matching that of the load to minimize the reflection of signals. The Multisim RF characterizer analysis helps you to study these impedances and choose the most appropriate frequency of operation.

Complete the following steps to use the simulator in order to read the desired variable:

1. Connect the **Network Analyzer** to the amplifier.
2. Run the simulator. Ignore any DC warnings and wait until the AC analyses are complete.
3. Double-click on the **Network Analyzer**.
4. Click **RF Characterizer** in the **Mode** box.
5. Under the **Trace** options, set the desired variable, from among **PG**, **APG**, and **TPG**. While the curves are plotted versus frequency, the numerical values are displayed at the top of curves for each frequency point. Refer to the *Power Gains* section for more information.
6. From the **Param.** drop-down list, select **Gains**. The voltage gain (**VG**) is plotted versus frequency and its value is given at the top of the curve.

Tip Use **Auto Scale** each time you change the parameters to get a better reading.

7. From the **Param.** drop-down list, select **Impedance**. The input/output impedances are provided in the form of a curve as well as printed out at the top of the curves.

8. Use the frequency scroll bar at the bottom of the curves to select the desired frequency for a specific variable.

Power Gains

The Multisim RF Simulator calculates the General Power Gain (**PG**), Available Power Gain (**APG**) and Transducer Power Gain (**TPG**) for Zo = 50 ohms at a given frequency. The dBMag is derived as $10log_{10}|PG|$.

PG is defined as the ratio of the power delivered to the load and the average power delivered to the network from the input, and is given as $PG = |S21|^2/(1-|S11|^2)$.

TPG is the ratio of the power delivered to the load to the power available from the source.

For Gs = GL= 0, TPG = $|S21|^2$.

APG is the ratio of the power available from the output port of the network to the power available from the source and it is expressed as

APG = $|S21|^2 / (1- |S22|^2)$

Voltage Gain

Voltage Gain, **VG**, is obtained for $\Gamma s = \Gamma_l = 0$ and is expressed as $VG = S21/(1 + S11)$.

Voltage Gain expressed in dBMag is calculated as $20log_{10}|VG|$.

If you observe the time domain signals of the input and output while the transistors are operating in the linear region, you find that the amplitude of the output voltage signal (when 50 Ohm load and source impedances are used) to the amplitude of the input voltage signal is the same as VG given by Multisim. Note, however, that VG is calculated using S-parameters.

Input/Output Impedances

These values are calculated assuming $\Gamma s = \Gamma_l = 0$. For this condition, we have:

$Zin = (1 + \Gamma in)/(1 - \Gamma in)$ where $\Gamma in = S11$ and

$Zout = (1 + \Gamma out)/(1 - \Gamma out)$ where $\Gamma out = S22$.

One must note that these values are normalized. The simulator prints denormalized values of Zin and Zout.

Matching Network Analysis

While designing RF amplifiers using Multisim, RF engineers need to analyze and, if necessary, modify circuit behavior. The **Matching Network Analysis** provides three options for analyzing circuit behavior:

- Stability circles
- Unilateral gain circles
- Impedance matching

These options are described in detail in this section. Depending on the application, one or more of the options is used. For example, to design oscillators only stability circles are used. On the other hand, to match an unconditionally stable circuit, the simulator first analyzes the stability properties of the circuit then uses automatic impedance matching.

The three options are accessed from the **Match Net. Designer** dialog box.

Note Before opening the **Match Net. Designer** dialog box, you must activate the circuit by clicking the **Run/resume Simulation** button and stop the simulation by clicking the **Stop Simulation** button.

Complete the following steps to open the **Match Net. Designer** dialog box:

1. Double-click the **Network Analyzer** on the circuit window.
2. Click the **Match Net. Designer** button. The **Match Net. Designer** dialog box appears.

Stability Circles

Stability circles are used to analyze the stability of a circuit at different frequency points.

In an ideal design, when an input signal is delivered to the input port of a two-port network, the entire source signal is delivered without any loss. In practice, however, part of the input signal bounces back to the source. Then, when the amplified signal is delivered to the load impedance, part of this signal bounces back to the output port of the amplifier. The amplifier, if it is not unilateral, transfers the reflected wave back to the source impedance. A circuit is considered unstable if the signal reflected is equal to the signal delivered in either the input or the output port.

An RF engineer aims to minimize this “bounce” effect and deliver maximum signal to the load. The stability circles in the **Network Analyzer** help achieve this goal.

Complete the following steps to perform the analysis:

1. Connect the biased amplifier to the **Network Analyzer** using two series capacitors (usually 100 F). The values of these capacitors are selected to minimize the numerical errors. In practice, however, two capacitors must be used to isolate the amplifier from the pre- and post-stage amplifiers in DC mode. Note that the impedance of these capacitors should not contribute to the attenuation of the input or output signal. The impedance of a capacitor is frequency dependant and is calculated using X_c = 1/(jwC) where "w" is (2*pi*f).
2. To activate the **Network Analyzer**, click **Run/resume simulation**. Wait until the AC analyses are complete. Ignore any warning for DC analysis.
3. Double-click on the **Network Analyzer** icon on the circuit window.
4. Click the **Match Net. Designer** button.
5. From the **Match Net. Designer** dialog box, click the **Stability Circles** tab.
6. Set the desired operating frequency in the **Freq.** field in the lower left corner of the dialog box, and click **OK**.

The result is a Smith Chart showing an input stability circle and an output stability circle. A stability circle represents the boundary between the values of source or load impedance that cause instability and those that do not. The perimeter of the circle thus represents the locus of points which forces K=1. Note that either the inside or the outside of the circle may represent an unstable region. Unstable regions are hatched on the Smith Chart.

There are three possible scenarios on the Smith Chart, as described below:

- **None of the Smith Chart is hatched** In this case the circuit is said to be "unconditionally stable", meaning that any area of the Smith Chart represents a valid passive source or load impedance. The designer can, then, select the input or output impedances using other criteria (such as gain or noise criteria).
- **Parts of the Smith Chart are hatched** In this case the circuit is "potentially unstable", meaning it is possible to select passive input or output impedance and still maintain the stability of the circuit. An input impedance should fall outside the hashed area of the input stability circle to achieve stability at the input port, while an output impedance should be selected outside the output stability circle to achieve stability at the output port.

- **The entire Smith Chart is hatched** In this case, the circuit is unstable regardless of input or output impedances. The designer has a number of options to achieve stability, including changing the frequency of operation, changing the DC biasing of the transistor, changing the transistor itself, or changing the entire structure of the amplifier.

In addition to stability circles, there are two numerical values printed on the **Match Net. Designer** dialog box. They are Delta and "K". The design is unconditionally stable if (|Delta| < 1) and K>1. For K<1, for example, the circuit is potentially unstable, and will most likely oscillate with certain combinations of source or load impedance.

Unilateral Gain Circles

This option is used to analyze the unilateral property of a circuit. A transistor is said to be unilateral when there is no "bounce" effect, meaning the signal reflected from the output port to the input port is zero.

This occurs if the reverse transmission coefficient, S_{12} or the reverse transducer power gain, $|S_{12}|^2$ is equal to 0. This means that the input section of the amplifier is completely isolated from the output section. (Note that passive networks are usually not unilateral.)

The unilateral property of a network is determined by calculating the **Unilateral Figure of Merit (U)**. If necessary, the frequency can be adjusted to improve the unilateral property.

Complete the following steps to calculate the **Unilateral Figure of Merit**:

1. From the **Match Net. Designer** dialog box click the **Unilateral Gain Circles** tab.
2. Read the value of "U" or the "Unilateral Figure of Merit".
3. Calculate the upper and lower limits of the following inequality using "U". $1/(1 + U)^2 < G_T/G_{TU} < 1/(1-U)^2$ where, G_T – transducer power gain, is defined as the ratio of the output power delivered to a load by a source and the maximum power available from the source, and $\mathbf{G_{TU}}$ represents the transducer power gain assuming unilateral property (S12=0) for the network. You need not calculate GT or GTU since only the limits are of interest here. If the limits are close to one, or "U" is close to zero, the effect of S12 is small enough to assume unilateral property for the amplifier. If it is not, go to the next step.
4. Change the frequency so that the minimum "U" is read. This frequency suggests an operating point for the amplifier where the unilateral property is best met.

Note The operating frequency to achieve the best unilateral property for the amplifier does not necessarily coincide with the maximum gain for the circuit. The unilateral gain circles are developed to identify the best load and source impedances to minimize the error due to unilateral assumption and maintain a satisfactory level of gain. The gain circles are also used for a trade off between the gain and the bandwidth.

Wide Band Amplifier

Complete the following steps to design a wide band amplifier, first find the maximum gain delivered by the circuit:

1. Double-click the **Network Analyzer** icon on the circuit window.
2. Click on the **RF characterizer** button.
3. Read the value of **TPG** (transducer power gain or G_T). This value is printed in dB.
4. Click on the **Match Net. Designer** button.
5. From the **Match Net. Designer** dialog box, click the **Unilateral Gain Circles** tab.
6. Change the value of **GS** and **GL** manually and individually until the circles become a dot on the Smith Chart.
7. Calculate the maximum transferable power.

   Gmax = Gs (dB) + TPG (dB) + GL(dB)

The maximum gain is achieved only for a narrow band. Since the slightest change in the circuit component would change its performance, the maximum gain is not achievable in reality. For a wider band of frequency, use a gain of less than the maximum.

Knowing the level of gain you want to achieve, complete the following steps toselect input and output impedances:

1. Select the desired gain (should be less than the maximum gain calculated above).
2. Choose **GS** and **GL** so that these three conditions are met:
   - GS + GTU + GL < Gmax
   - GS < Gsmax
   - GL < Glmax.
3. Enter the selected values for **GS** and **GL** and observe the circles. Select points on the circles closest to the center of the Smith Chart. These points are shown on the Smith Chart and circles by two triangles. The circle for GL = 0 dB always passes through the center. Therefore, the best point of the GL to produce GL = 0 dB.

Any point selected on the Smith Chart is a normalized point. These points provide the impedances for the input and output ports that you then design manually.

Note To ensure these points will not cause instability, we recommend you follow the instructions in the *Stability Circles* section. If the amplifier is "unconditionally stable", it would be stable for any passive load or source network, so you need not check stability circles in this case.

Impedance Matching

Occasionally, a design is considered "unconditionally stable", meaning the amplifier does not oscillate in the presence of any passive load or source impedance. In this case, you can use the impedance matching option to automatically modify the structure of an RF amplifier to achieve maximum gain impedance.

To deliver maximum power, a circuit must match at both its input and output ports. In other words, there needs to be maximum matching between the output of the amplifier and the output impedance, and the input of the amplifier and the source impedance. There are eight possible structures for each port, although only a few of these provide complete matching.

Complete the following steps to use impedance matching to find a matching network:

1. Connect the **Network Analyzer** to your amplifier as described in the *Stability Circles* section. Do not forget the capacitors in order to prevent DC loading of the **Network Analyzer**.
2. Run the simulation.
3. Double-click the **Network Analyzer** icon.
4. Click the **Match Net. Designer** button.
5. On the **Match Net. Designer** dialog box, click the **Impedance Matching** tab.
6. Change the frequency to the desired operating point.
7. Enable the **Auto Match** checkbox.

The instrument provides the structure as well the numerical values of components. You can click on the left and right sides of the **Impedance Matching** dialog box to change the structure. However, only a few of the eight structures can provide matching.

Noise Figure Analysis

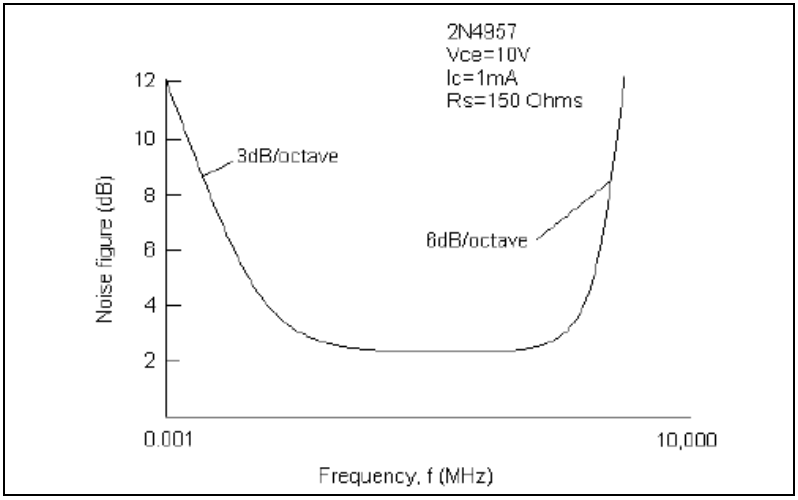
The noise figure is used to specify exactly how noisy a device is. For a transistor, noise figure is simply a measure of how much noise the transistor adds to the signal during the amplification process.

In a circuit network, the noise figure is used as a “figure-of-merit” to compare the noise in a network with the noise in an ideal or noiseless network. It is a measure of the degradation in signal-to-noise ratio (SNR) between the input and output ports of a network.

When calculating the noise figure of a circuit design, **Noise Factor (F)** must also be determined. This is the numerical ratio of noise figure, where noise figure is expressed in dB. Thus,

Noise Figure = 10log10F
And,
F = Input SNR
Output SNR

The noise figure for any device is provided on the part datasheet and is plotted similar to the graph below (for the 2N4957):



Another method of describing noise figure is expressed by the following equation:

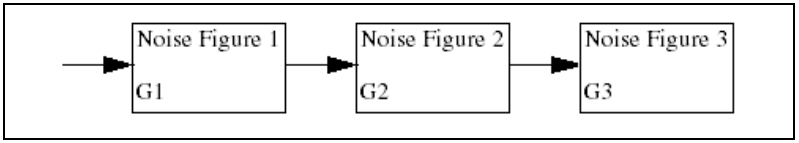
$$NoiseFigure = 10\log10\frac{Si/Ni}{So/No} = 10\log10NR$$

Where *Si/Ni* is the signal-to-noise power ratio at the device's input and *So/No* is the signal-to-noise power ratio at its output. The term (*Si/Ni, So/No*) is the *noise ratio* (NR). If the device under consideration is ideal (injected no additional noise), then *Si/Ni* and *So/No* are equal—in this case, NR = 1, and Noise Figure = 10log(1) = 0 dB.

Note This result cannot be obtained in practice.

Cascaded Networks

The following is an example of a cascade network:



Often, it is necessary to calculate the noise figure of a group of amplifiers that are connected in cascade. This is easily done if the noise figure of each individual amplifier in the cascade configuration is known.

$$F_{total} = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \frac{F_4 - 1}{G_1 G_2 G_3}$$

where,

F_n = the noise factor of each stage

G_n = the numerical gain of each stage (not in dB)

Another look at the above equation reveals an interesting point. If the gain of the first stage is sufficiently high, the denominators of the second and succeeding terms will also be very large (since G1 is multiplied with the denominator of every term). This will force all other terms to be very small values, leaving only F1 in the equation. Hence, the noise figure of the first stage will typically determine the noise figure of the cascade network.

Noise Figure in SPICE Models

In order to understand how **Noise Figure Analysis** operates within Multisim, you must first understand how noise figure is introduced to a SPICE model.

Every simulatable part in a Multisim circuit design contains a SPICE model. This model describes how the part functions in simulation. There are two possible methods of constructing a model:

- Create a SPICE circuit that emulates the function of the desired part (for example, an Op-amp model in SPICE is constructed using several transistors, resistors, etc).
- Certain components are already defined in a SPICE engine (for example, transistors). Models for these components are simply a list of parameter values (for example, Beta) that feed into a pre-defined circuit already included in the SPICE engine.

This example analyzes option 2.

Below is a SPICE model for a typical transistor:

```
.MODEL BF517 NPN (IS=0.480F NF=1.008 BF=99.655 VAF=90.000
IKF=0.190
+ ISE=7.490F NE=1.762 NR=1.010 BR=38.400 VAR=7.000
IKR=93.200M
+ ISC=0.200F NC=1.042
+ RB=1.500 IRB=0.100M RBM=1.200
+ RE=0.500 RC=2.680
+ CJE=1.325P VJE=0.700 MJE=0.220 FC=0.890
+ CJC=1.050P VJC=0.610 MJC=0.240 XCJC=0.400
+ TF=56.940P TR=1.000N PTF=21.000
+ XTF=68.398 VTF=0.600 ITF=0.700
+ XTB=1.600 EG=1.110 XTI=3.000
+ KF=1.000F AF=1.000)
```

Although noise figure in a model is achieved using a combination of several model parameters, parameter NF is the most significant value (when simulating noise figure in a transistor model). When selecting a SPICE model to use for **Noise Figure Analysis**, you must ensure that this parameter exists. If not, the part will operate without noise parameters and therefore will generate no noise.

Refer to the *BJT* section for a description of the parameters in the above SPICE model.

Multisim Approach

Multisim calculates the noise figure using the equation:

$$F = \frac{N_o}{GN_s}$$

Where N_o is the output noise power (which includes the noise due to two-part network and the magnified input noise) N_s is the thermal noise of the source resistor (this resistor generates noise equal to the output noise of the previous stage), and G is the AC gain of the circuit (the ratio of the output signal to the input signal of the two-part network). Note that the bandwidth of the signal has been considered in the source resistor.

Finally, Multisim prints the Noise Figure in dB, that is 10 log10 (F).

Noise Figure Analysis Tabs

Just as for other Multisim analyses, you must fill in the appropriate fields in the **Analysis Parameters** tab.

Setting Noise Figure Analysis Parameters for Normal Use

For normal use, you only need to:

- select an input source from the **Input noise reference source** drop-down list
- select an output node from the **Output node** drop-down list
- select a reference node from the **Reference node** drop-down list
- enter a value in the **Frequency** field
- enter a value in the **Temperature** field.

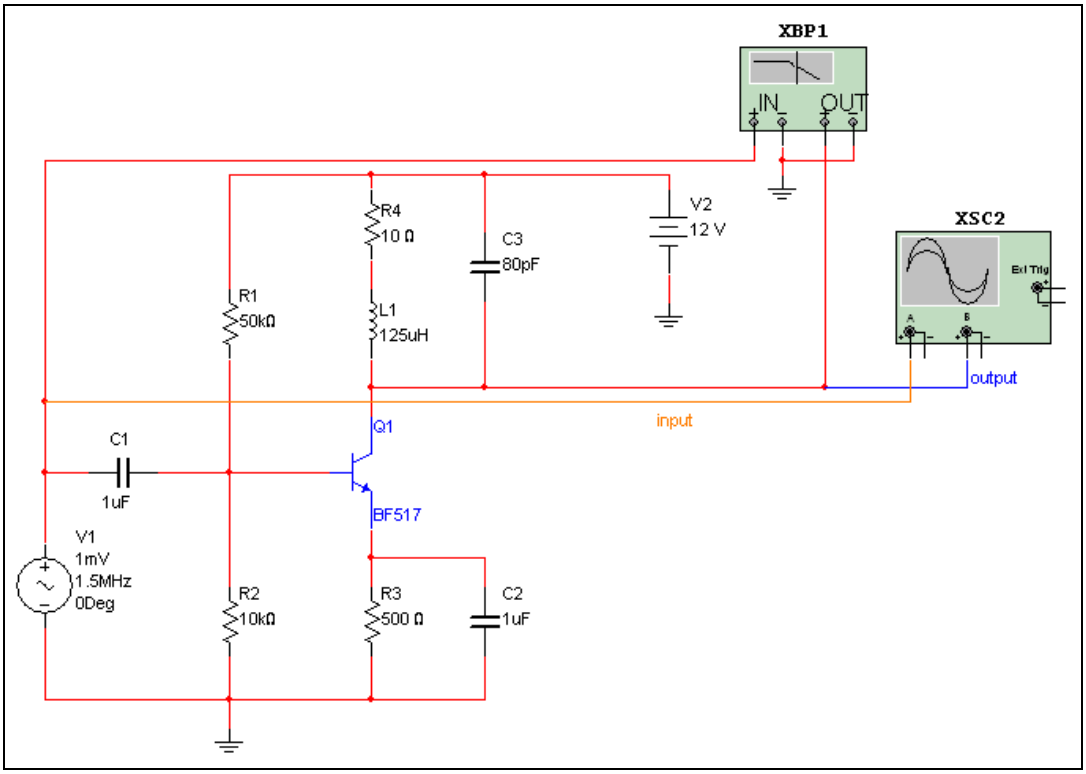
You can filter the variables displayed to include internal nodes (such as nodes inside a BJT model or inside a SPICE subcircuits), open pins, as well as output variables from any submodules contained in the circuit.

Complete the following steps to filter the variables displayed:

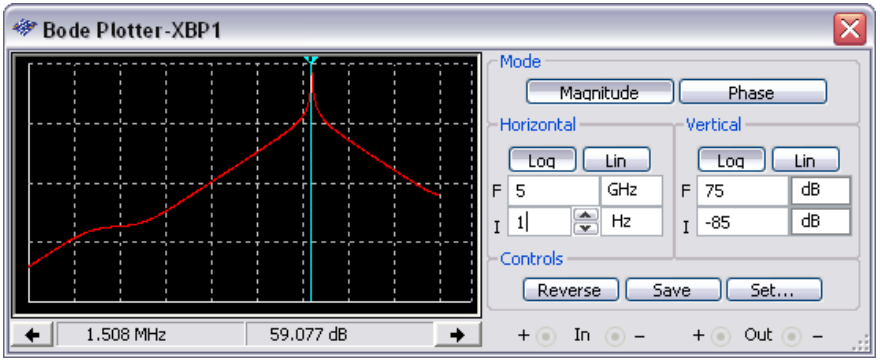
1. Click the **Change Filter** button. The **Filter Nodes** dialog box appears.
2. Enable one or more settings.
3. Click **OK**.

Sample Noise Figure Analysis

To illustrate how **Noise Figure Analysis** operates, this example uses the following circuit:



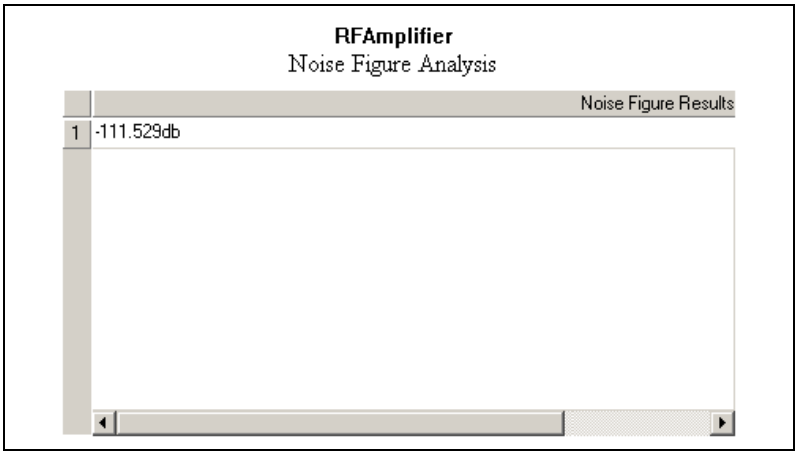
The circuit frequency response is shown below:



Notice that the maximum gain appears at approximately 1.5MHz.

Complete the following steps to set up **Noise Figure Analysis** using the circuit shown:

1. Select **Simulate»Analyses»Noise Figure Analysis**.
2. Under the **Analysis Parameters** tab, set the following:
   - **Input noise reference source** to vv1 (which is the input AC Source)
   - **Output node** to V(output)
   - **Reference node** to V(0)
   - **Frequency** to 1.5e+006 (1.5 MHz)
   - **Temperature** to 27 degrees C
3. Click **Simulate**. The following displays.



Therefore, the **Noise Figure** for the circuit is approximately –110 dB.

RF Model Makers

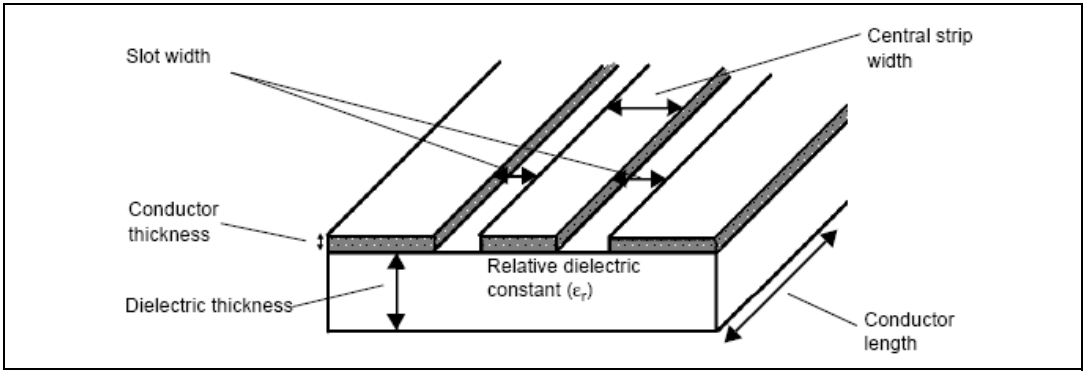
As with the other Multisim Model Makers, RF Model Makers automatically simulate models based on the input you provide. Whereas input for other model makers usually comes from data books, RF Model Makers can also receive other types of input, such as operating characteristics or physical dimensions, depending on the type of components you are modelling.

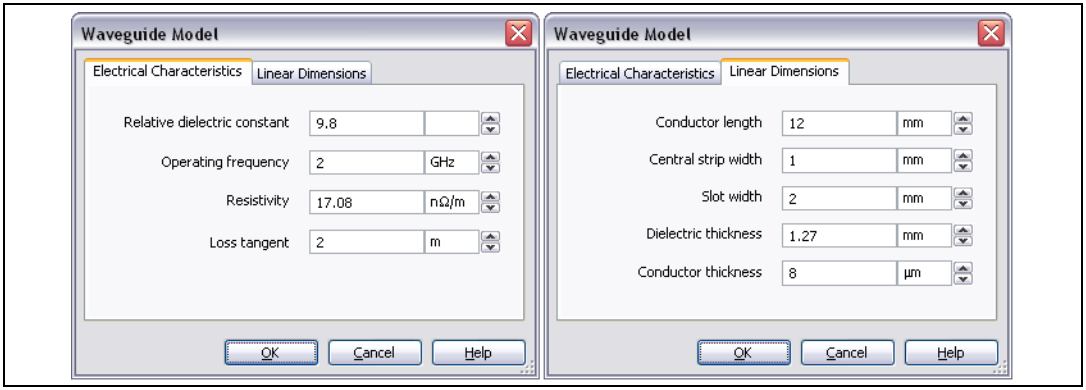
Multisim has RF Model Makers for the following types of components:

- *Waveguide*
- *Microstrip Line*
- *Open End Microstrip Line*
- *RF Spiral Inductor*
- *Strip Line Model*
- *Stripline Bend*
- *Lossy Line*
- *Interdigital Capacitor*.

Waveguide

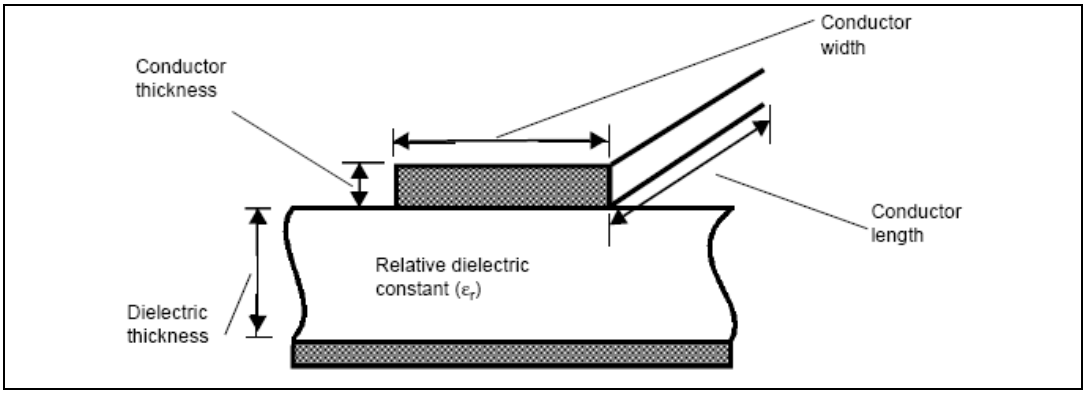
For Waveguide models, enter values in the two tabs shown below. Use the following diagram for assistance in identifying the values:

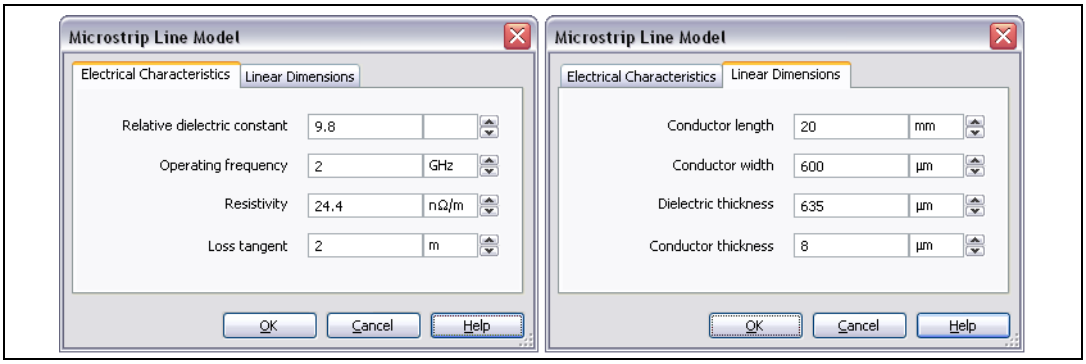




Microstrip Line

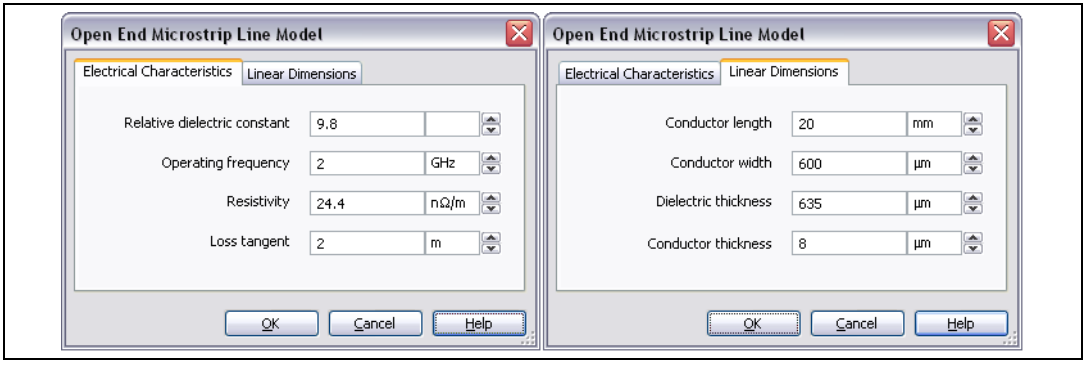
For the Generic Microstrip Line models, enter values in the two tabs shown below. Use the following diagram for assistance in identifying the values:





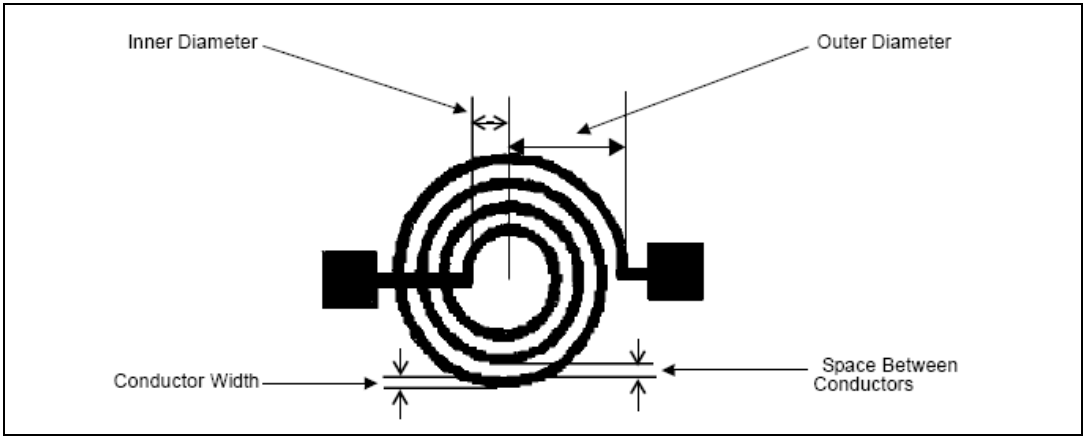
Open End Microstrip Line

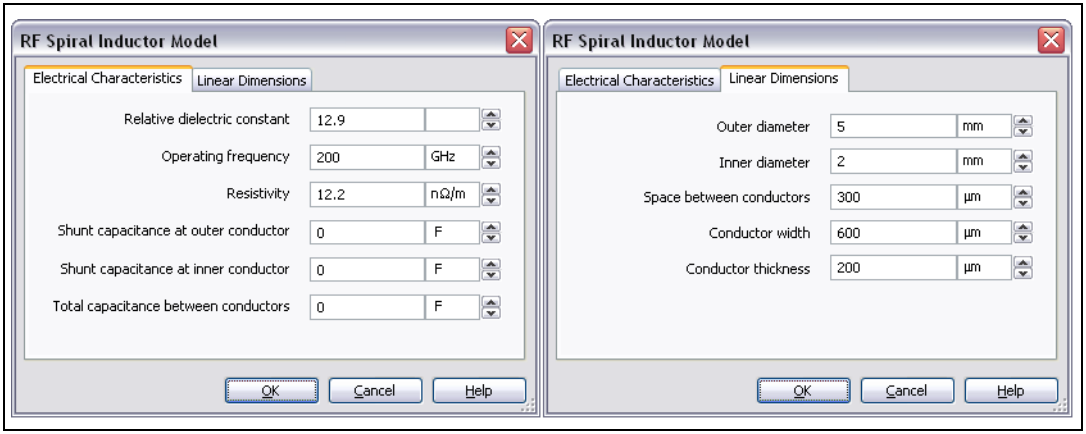
For Open End Microstrip models, enter values in the following tabs:



RF Spiral Inductor

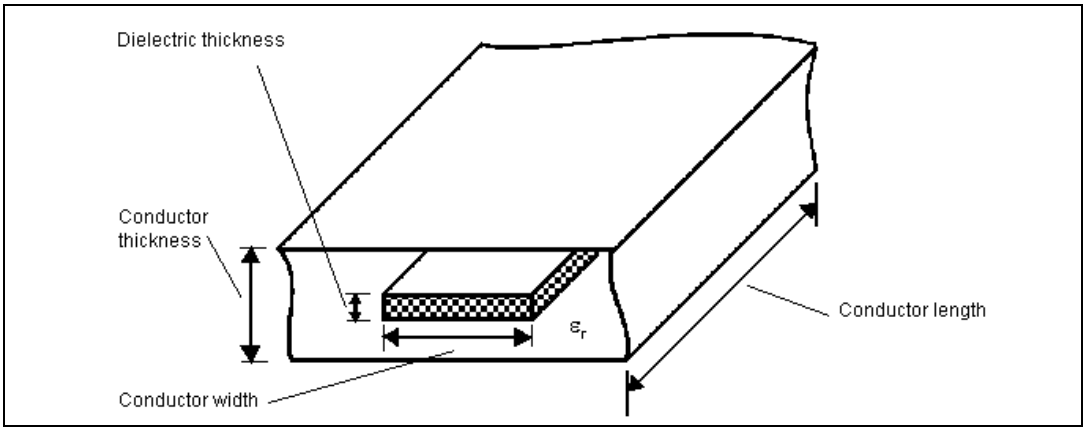
For the RF Spiral Inductor models, enter values in the two tabs shown below. Use the following diagram for assistance in identifying the values:

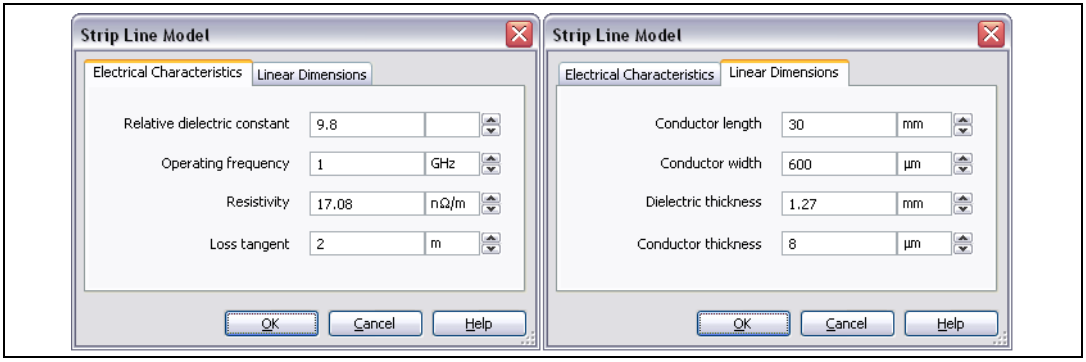




Strip Line Model

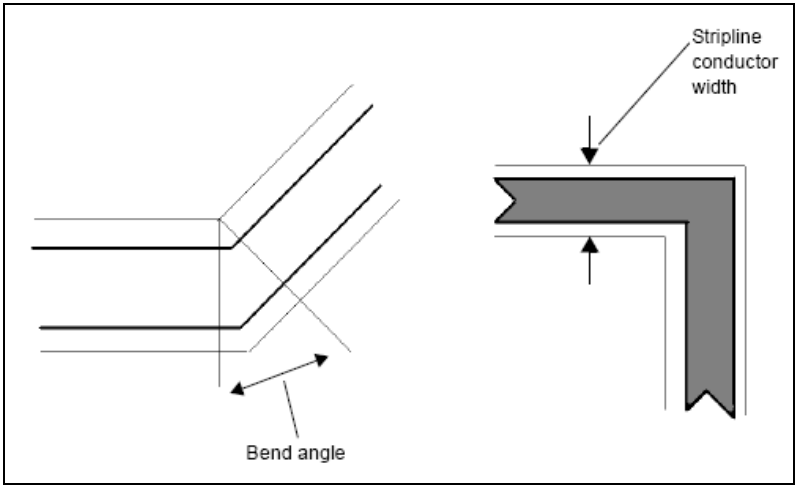
For the Strip Line models, enter values in the two tabs shown below. Use the following diagram for assistance in identifying the values:

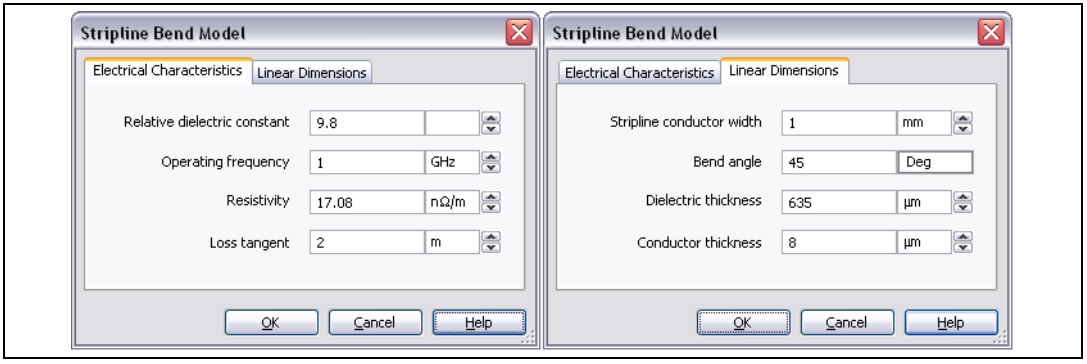




Stripline Bend

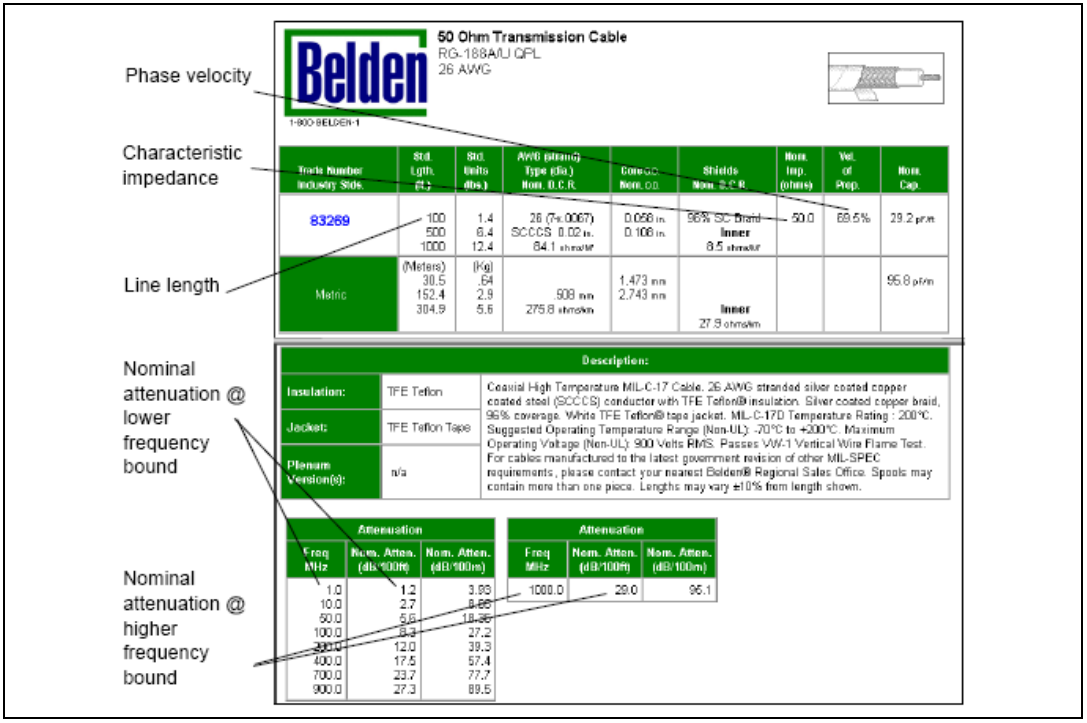
For the Stripline Bend models, enter values in the two tabs shown below. Use the following diagram for assistance in identifying the values (refer to the Stripline diagram in the *Strip Line Model* section for Relative dielectric constance (ε_r), Dielectric thickness and Conductor thickness):

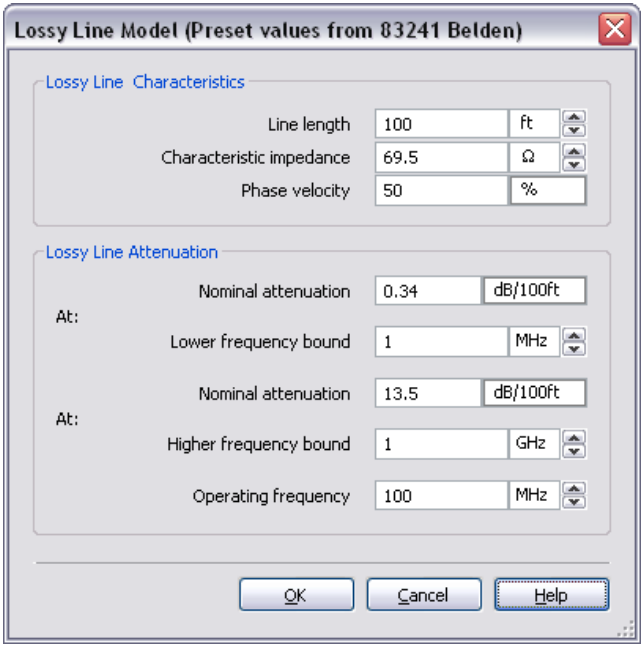




Lossy Line

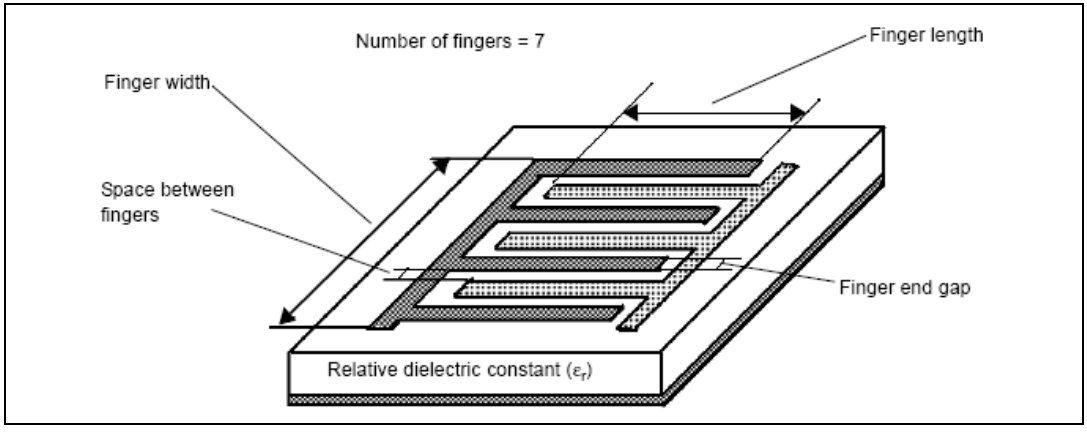
For Lossy Line models, enter values in the dialog shown below. Use the following catalogue excerpt for assistance in identifying the values:

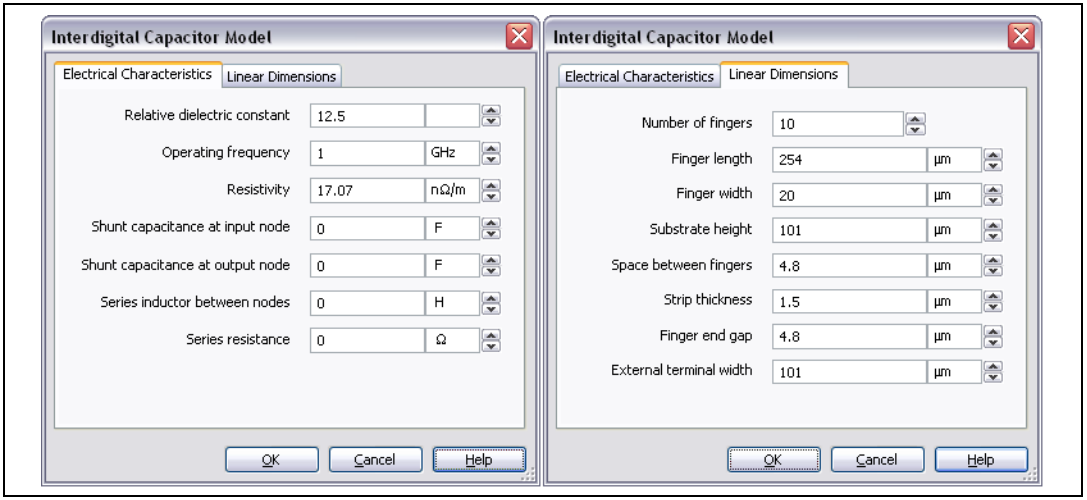

Phase velocity
Characteristic impedance
Line length
Nominal attenuation @ lower frequency bound
Nominal attenuation @ higher frequency bound
Belden
50 Ohm Transmission Cable
RG-188A/U CPL
26 AWG
1-800-BELDEN-1
83269
Description:
Insulation:
TFE Teflon
Jacket:
TFE Teflon Tape
Plenum Version(s):
n/a
Coaxial High Temperature MIL-C-17 Cable, 26 AWG stranded silver coated copper coated steel (SCCCS) conductor with TFE Teflon® insulation. Silver coated copper braid, 96% coverage. White TFE Teflon® tape jacket. MIL-C-17D Temperature Rating : 200°C. Suggested Operating Temperature Range (Non-UL): -70°C to +200°C. Maximum Operating Voltage (Non-UL): 900 Volts RMS. Passes VW-1 Vertical Wire Flame Test. For cables manufactured to the latest government revision of other MIL-SPEC requirements, please contact your nearest Belden® Regional Sales Office. Spools may contain more than one piece. Lengths may vary ±10% from length shown.
Attenuation
Freq MHz
Nom. Atten. (dB/100ft)
Nom. Atten. (dB/100m)
1.0 1.2 3.93
10.0 2.7 8.86
50.0 5.6 18.36
100.0 8.3 27.2
200.0 12.0 39.3
400.0 17.5 57.4
700.0 23.7 77.7
900.0 27.3 89.5
Attenuation
Freq MHz
Nom. Atten. (dB/100ft)
Nom. Atten. (dB/100m)
1000.0 29.0 95.1



Interdigital Capacitor

For Interdigital Capacitor models, enter values in the two tabs shown below. Use the following diagram for assistance in identifying the values:





Tutorial: Designing RF Circuits

This tutorial:

- provides an introduction to simple RF circuit design
- demonstrates to engineers how to use Multisim for designing an RF circuit. Each design step is accompanied by the required simulation steps in Multisim.

The methodology that an RF engineer uses to design an RF circuit differs from that used for a low-frequency circuit design. An RF designer looks at performance parameters such as S-parameters, input/output impedances, power gain, noise figure, and stability factor. These design parameters are not directly available from a SPICE simulation. Impedance matching is a phase of RF circuit designs where the designer uses a Smith Chart, and calculates the values of matching elements such that maximum power is transferred to the load impedance. The Smith Chart or the calculations are not provided by SPICE simulation.

Selecting Type of RF Amplifier

Select the type of amplifier based on the application. Amplifiers designed for low-power applications are different than those for low-noise applications. Similarly, broad-band amplifiers are different in terms of design and structure than those for high-gain amplifiers. Some of the possible applications are:

- Maximum Power Transfer—These types of amplifiers operate in a very narrow band of frequencies.
- Design for Specified Gain—You may intentionally introduce mismatching at the input and/or the output ports to improve the bandwidth, even though the resulting power transfer is not maximal.
- Low-Noise Amplifier Design—In receiver applications, you need a pre-amplifier with as low a figure noise as possible since the first stage of a receiver front end has a dominant effect on the noise performance of the overall system. It is not possible to obtain both minimum noise figure and maximum gain for an amplifier.
- Oscillators—To produce a sinusoidal steady-state RF signal, you can use active elements and intentionally introduce negative resistance.

The **Network Analyzer** provided by Multisim is not intended for high-power RF amplifiers, because the **Network Analyzer** performs small-signal analyses of the RF network only.

In this tutorial, we will design the amplifier for maximum power transfer. Then, you will provide design steps for constant gains.

To prepare for the tutorial, open a new circuit window.

Selecting an RF Transistor

Different types of transistors are designed for a broad range of applications. For example, you may find many transistors for low noise applications. Cost of the transistor plays a significant role in choosing from those offered. In this tutorial, we will choose MRF927T1 because it is used for low power, low noise applications at relatively high frequencies.

Complete the following steps to select the MRF927T1 transistor in Multisim:

1. Click the **RF** button in the **Components** toolbar.

2. Select the **RF_BJT_NPN** family in the **Select a Component** browser.
3. Scroll down in the component list until you find MRF927T1 and select it. Component data for that component appears in the dialog box.

4. Click **OK**. The browser closes and your cursor changes to indicate a transistor is ready to be placed.
5. Click to place the transistor on the circuit window.

Selecting a DC Operating Point

DC Operating point is referred to as Vce and Ic. There are many reasons to select a specific DC Operating point. You need to consider “maximum swing” at the output, small/portable power source, and gain-bandwidth. Some DC Operating points are available in data books, and others must be decided based on the application.

Vce Settings

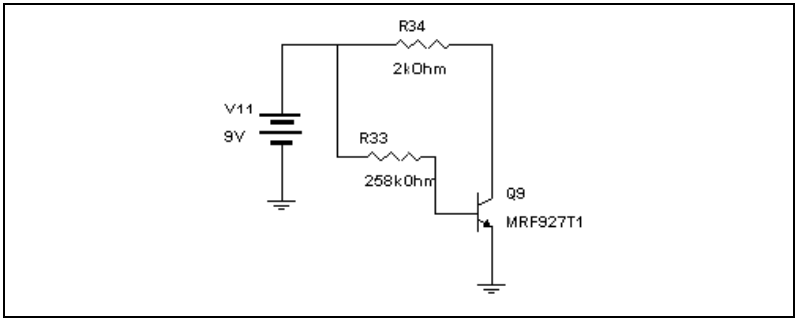
Vce is always less than Vcc, and it is usually around Vcc/2 for maximum swing in a common-emitter configuration. For this tutorial Vce=3V and Vcc=9V are selected.

Ic Settings

The nominal value of Ic for the selected transistor is 5mA. The power dissipated in the transistor at any time is Ic*Vce. For this tutorial, Ic=3mA is selected to dissipate less power and to be close to the nominal value of Ic. This will let us achieve relatively good current-gain bandwidth and moderate voltage gain. (The voltage gain is maximum at Ic=1mA and the current gain-bandwidth is maximum at Ic=5mA.)

Selecting the Biasing Network

There are a number of possible structures to select from for proper DC biasing of the network. It is important to note that the performance of the transistor and the amplifiers depends on DC Operating points. The following figure shows one possible biasing network.



This is the simplest structure for a biasing network. However, its thermal stability is poor. To find the resistor values for this structure you need to know five values: Vce, Ic, Vcc, Vbe, and Beta (which is the DC current gain of the transistor and is given in most data books). Beta relates Ic to Ib as Beta = Ic/Ib. Vbe is the base-emitter voltage of the transistor when it is active, and is typically 0.7 V. Both Beta and Vbe depend on the values of Ic and Ib. The initial design process starts with typical values of Rc and Rb in the structure shown above. However, if accuracy is critical, you should use Multisim to ensure that the values of Ic and Vce are as intended. In this tutorial, these selections are used:

- Vce = 3 V
- Ic = 3 mA
- Vcc = 9 V
- Vbe = 0.7 V
- Beta = 100.

The initial values of Rc and Rb are calculated as shown below.

- Rc = (Vcc-Vce)/Ic = (9 V - 3 V)/ 3 mA = 2 kOhm
- Ib = Ic/Beta = 3 mA/100 = 30 µA
- Rb = (Vcc - Vbe)/Ib = (9 V - 0.7 V)/30 µA = 277 kOhm

Complete the following steps to select the DC operating points:

1. Draw the circuit shown above with Rb=277 kOhm and Rc=2 kOhm. Note that Vcc=9 V and the transistor is MRF927T1.
2. Click the **Analysis** button and choose **DC Operating Point**. The **DC Operating Point Analysis** dialog box appears.
3. Select the nodes representing the collector and base of the transistor.
4. Click **Simulate**.

The results will show a reading of Vc and Vb. In the DC biasing network, Vc is Vce and Vb is Vbe. The first readings are Vce=3.33 V and Vbe=0.8 V. You can modify the value of either Rc or Rb or both, to achieve the desired DC operating point. After a number of iterations, you arrive at Rc=2 kOhm and Rb=258 kOhm. Reading the values of Vce and Vbe for the final simulation, you will note that

$$\text{Beta} = \text{Ic/Ib} = \text{Rb}*(\text{Vcc-Vce}) / [\text{Rc}. (\text{Vcc-Vbe})] = 94.36$$

which is close to the initial value of Beta.

Selecting an Operating Frequency Point

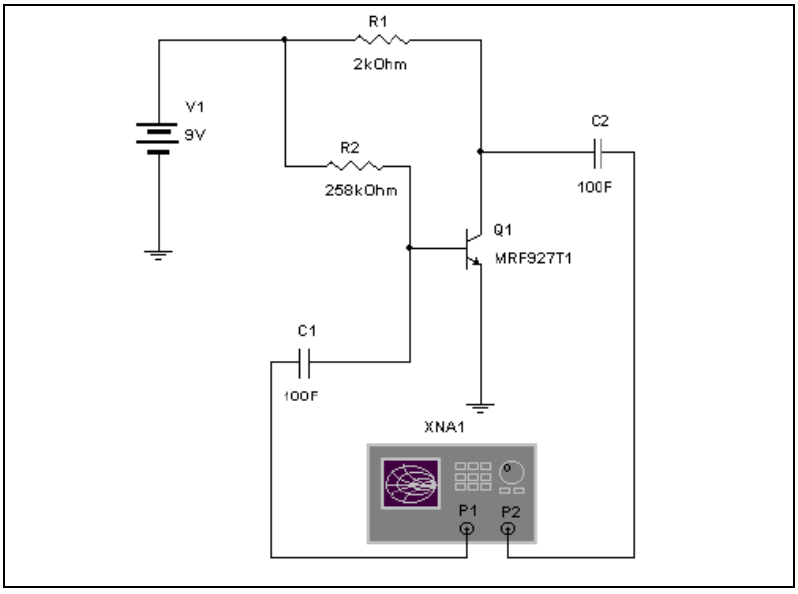
The operating frequency point chosen depends on the type of application, and is usually defined in the design specifications. For this tutorial, you can assume a single (center) frequency analysis of 3.02 GHz.

Analyzing the RF Network

Complete the following steps to perform the simulation:

1. Connect the biased transistor to the **Network Analyzer** using two series capacitors. These capacitors are used to isolate the **Network Analyzer** from the biasing network in DC mode. This step is necessary whenever the biasing network is important, that is, for active circuits only.

   The connection should look like this:

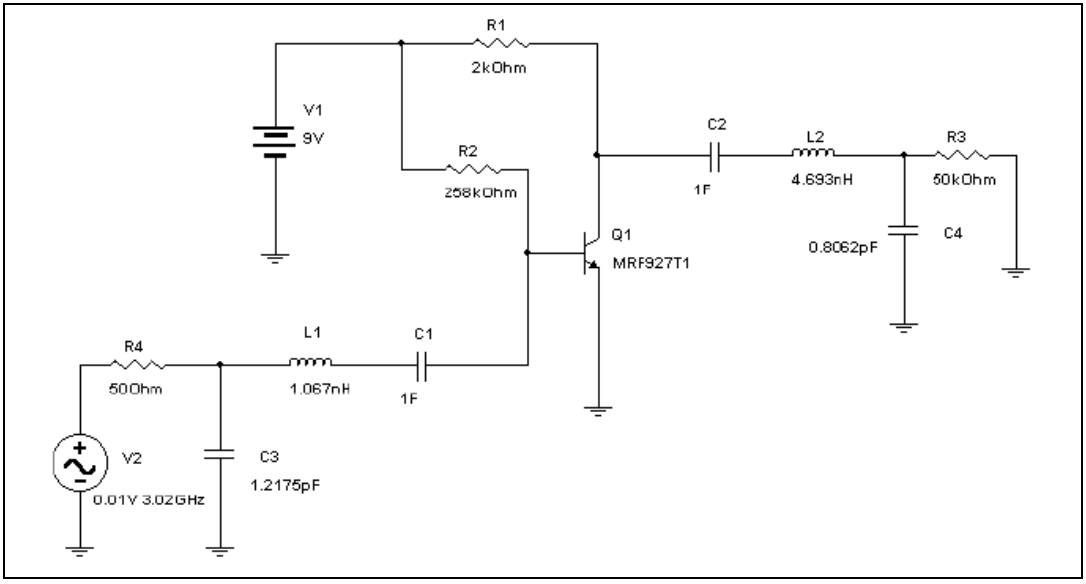


2. Choose **Simulate»Run** and wait until the AC analyses are complete. Ignore any warnings for DC analyses.
3. Double-click on the **Network Analyzer** icon and click the **Match Net. Designer** button.
4. In the **Match Net. Designer** dialog box that appears, do the following:
   - set the frequency to 3.02 GHz

- since the circuit is “unconditionally stable” for this frequency point, click **Impedance Matching**
- since the circuit is “unconditionally stable”, automatic impedance matching is possible. Enable **Auto Match**.

The dialog box provides the structure and the values necessary for conjugate matching. Hence, maximum power transfer is achieved.

Below is our design for maximum power transfer at f=3.02 GHz:



Note 1F is needed to isolate the active network from its matching network to keep the transistor in its biasing state.

Impedance matching yields maximum power transfer for a very narrow bandwidth. In real applications, however, you need to balance the power transfer and the bandwidth. For this reason, mismatching is intentionally introduced to the circuit.

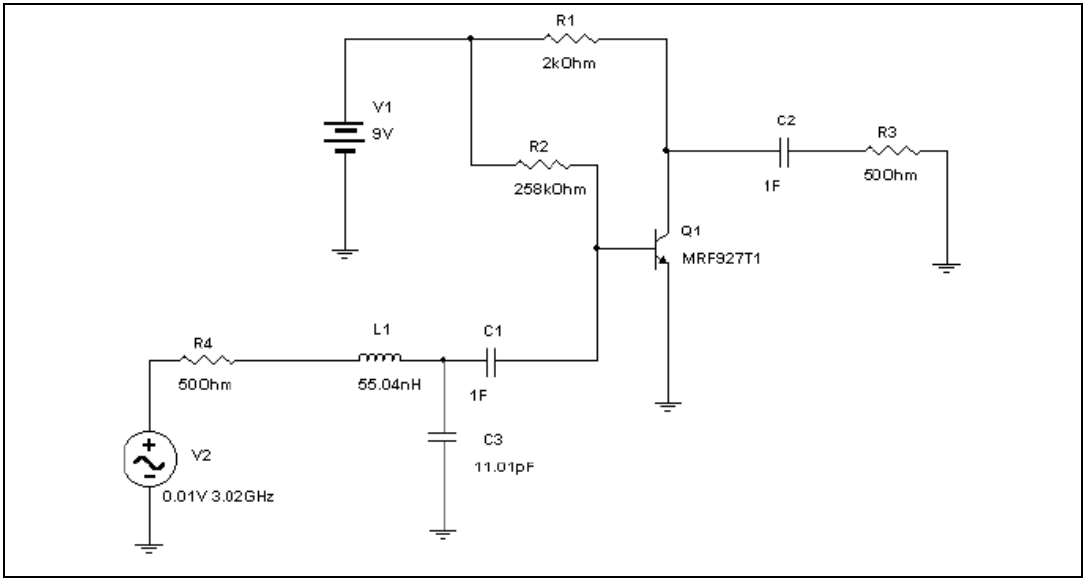
Complete the following steps to design the amplifier for a constant gain less than the maximum possible gain, for better frequency response:

1. Open the DC biased transistor circuit discussed previously.
2. Open the **Network Analyzer** and change settings as follows:
   - click **RF Characterizer**

- select “Power Gains” from the **Param** drop-down list
- select “dB MAG” from the **Marker** drop-down list
- set the frequency to 3.02 GHz
- click **TPG** (Transducer Power Gain) under **Trace** and set its value to 4.3652 dB for our example.

3. Click **Match Net. Designer** and, in the **Match Net. Designer** dialog box, select the Unilateral Gain Circles tab.
4. Change the value of Gs and Gl manually and individually until the circles become a dot on the Smith Chart.The example shows Gs=0.042 dB and Gl=1.2650 dB.
5. Calculate the maximum transferable power. Pmax = 0.042 + 4.3652 + 1.2650 = 5.6722 dB.
6. Select the gain desired. This gain should be less than 5.6752 dB. 3.5302 dB was selected as the power gain.
7. Choose Gs and Gl so that Gs + 4.3652 + Gl = 3.5302 dB and Gs < 0.042 dB and Gl < 1.2650 dB. You selected Gs = -0.08350 dB and Gl = 0 dB.
8. Enter the selected values for Gs and Gl and observe the circles. Select point or points on the circles which are closest to the center of the Smith Chart. The circle for Gl = 0 dB always passes through the center. Therefore, the best point of the Gl is the center itself. This means that 50 Ohm load is sufficient to produce Gl = 0 dB. The best point on Gs for the example is Zl = 2 (normalized). Using this value, you can design the matching network at the input port of the amplifier. You must make sure that the selected point or points are stable. Therefore, it is recommended that you go back to “Stability Circles”, and confirm the stability of the design. Since the amplifier is unconditionally stable, it would be stable for any passive load or source network. Hence, you need not check the stability circles.

The complete amplifier is shown in the following figure:



Note The matching elements are calculated manually in this example.

15

Multisim MCU

Multisim's MCU co-simulation system provides software development features for writing and debugging code for embedded devices.

Introduction to Multisim MCU

Microcontroller (MCU) components are useful for many circuit designs. A modern microcontroller typically combines a CPU, data memory, program memory, and peripheral devices on a single physical chip. The integration of these essential elements of a computer into a single chip reduces component counts and board size resulting in higher reliability with more capabilities.

Embedded software development can be a challenging process for even the best programmers. Multisim helps you produce high quality code more quickly and easily. The MCU development interfaces allow you to pause a simulation, inspect the internal memory and registers of the MCU, set code breakpoints and single step through your code.

Note Refer to the *Getting Started with NI Circuit Design Suite* manual for a tutorial covering Multisim's main MCU functions.

Multisim MCU Basics

This and subsequent sections give a brief overview of the Multisim's MCU functionality.

Complete the following steps to place an MCU:

1. Select **Place»Component** to display the **Select a Component** dialog box.
2. Navigate to the **MCU Group** and select the **Family** containing the desired MCU (for example, 805x, PIC).
3. Select the desired MCU, click **OK** and click again to place the component on the workspace. The **MCU Wizard** dialog appears. The **MCU Wizard** helps you get started by creating an MCU Workspace, default project and source file.

4. Complete the steps in the wizard as detailed in the *MCU Wizard* section.
5. Use the **MCU Code Manager** to manage the MCU files in your MCU workspace and to set the build settings for your MCU project. Refer to the *MCU Code Manager Overview* section for more information.

Debugging Tools

Multisim's MCU debugging tools give you the ability to control execution at the instruction level while also providing views of the memory and registers within the MCU.

After an MCU's project has been built (without errors) using the **Build** command, the debug window becomes accessible.

To open the debug window for a particular MCU, select **MCU»<MCU name, e.g., MCU 8051 U1>»Debug View**.

This opens a new window in Multisim with a drop-down list at the top used to select between the different views.

There is one view available for each source code file, as well as one for the disassembly of the entire project. By default, opening a **Debug View** opens the project disassembly view which is a complete disassembly of the ROM for the relevant MCU. In the project disassembly view, every memory address is shown from address zero to the maximum address of the ROM. The project disassembly view always shows the disassembly (not the listing assembly) because it shows memory addresses which may not have a corresponding listing (such as the opcodes from statically linked libraries).

Refer to the *MCU Debugging Features* section for more information.

MCU Source Code Editor

The source code view shows the assembly or C source code for the MCU program.

The numbers on the far left are the program memory addresses and the hexadecimal codes to their immediate right are the assembled codes for each mnemonic assembly instruction. The column of numbers in the middle shows the line number in the original assembly or C source. The remainder of the line to the right shows the assembly source and comments.

MCU Memory View

The contents of the **MCU Memory View** change depending on the type of MCU. It may, for example, contain internal memory information, register views and configuration information.

Refer to the *Component Reference help file* for information about specific MCUs.

Advanced Features

Multisim's MCU functionality includes advanced debugging tools that make it easy to pause your circuit and explore the internal data and state of the MCU controlling your circuit. You can set breakpoints and single step through assembly code while validating that the register contents are changing as expected. You can also manually edit most memory views while debugging.

An example of these advanced debugging features is found in *Getting Started with NI Circuit Design Suite*.

Peripheral Devices

Along with its selection of MCUs, Multisim contains a number of peripheral devices.

The **MCU Group** contains RAM and ROM devices that are designed to function specifically with the MCUs.

The **Advanced Peripherals Group** contains a selection of Keypads, LCDs, Terminals and Miscellaneous Peripherals such as the Liquid Holding Tank.

Refer to the *Component Reference help file* for more information.

Multisim MCU File Management

This section describes the file management features used in Multisim MCU.

MCU Design Overview

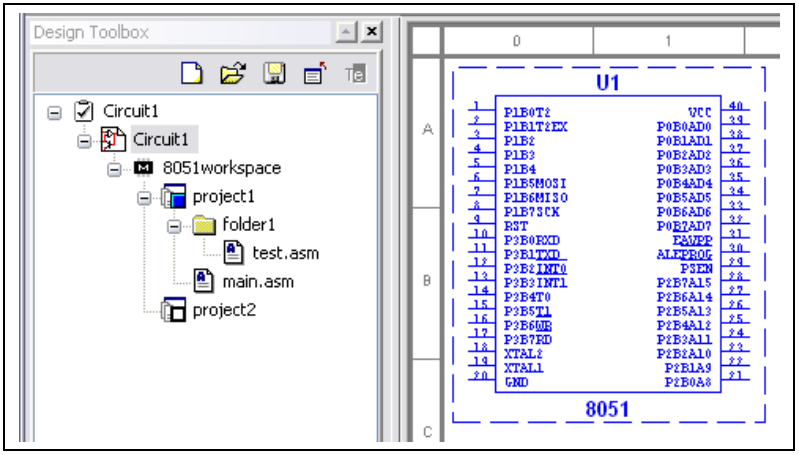
Every MCU component that is placed on the circuit schematic is associated with an MCU design consisting of a set of MCU files that exist on your hard disk separate from the Multisim circuit file (`*.ms10`):

MCU File Type	File Extension	Description
MCU workspace file	`.mcuws`	Maintains information on projects contained in the MCU workspace.
MCU project file	`.mcuprj`	Maintains information on files contained in the MCU project.
Assembly source file	`.asm`	Assembly code
Assembly include file	`.inc`	Assembly code
C source file	`.c`	C code
C header file	`.h`	C code

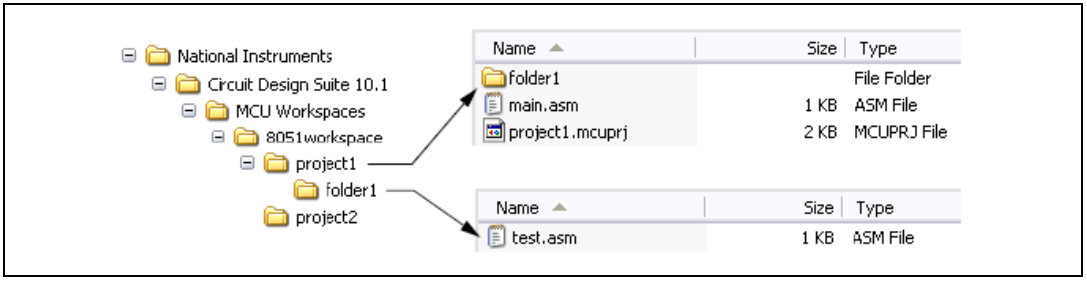
Each MCU workspace file is contained inside a workspace folder with the same name as the workspace file. All the other MCU-related files for this particular MCU component are stored inside the MCU workspace file in a hierarchical manner.

Each MCU project has its own project folder inside the workspace folder. Each MCU project file resides inside the project folder and the name of the project folder is the same as the project file. The assembly and C files are contained inside the project folder and also can be stored inside additional folders nested inside the project folder.

Below is an example of a circuit (`Circuit1`) in Multisim. It contains an 8051 MCU component (U1) that has an MCU design in the MCU workspace `8051workspace` as shown in the **Design Toolbox** view. The MCU workspace contains two projects, `project1` and `project2`. `Project1` has some MCU files, but `project2` is empty.



The MCU files for the above circuit are organized on disk in the file structure shown in the figure shown below. The arrows point to the contents contained in the project1 and folder1 folders.



Multisim stores the path to the MCU workspace file inside the circuit file (`*.ms10`). Every time `Circuit1.ms10` is opened, the MCU workspace must be found on disk in order for Multisim to load the MCU design and display it in the **Design Toolbox** or perform any actions on the MCU workspace.

MCU Wizard

The **MCU Wizard** helps you get started by creating an MCU Workspace, default project and source file.

Complete the following steps to invoke the **MCU Wizard**:

1. Select **Place»Component** to display the **Select a Component** dialog box.

2. Navigate to the **MCU Group** and select the **Family** containing the desired MCU (for example, 805x, PIC).
3. Select the desired MCU, click **OK** and click again to place the component on the workspace. Step 1 of the **MCU Wizard** dialog appears. This is where you specify the MCU workspace information.
4. Complete the following as desired:
   - **...workspace path for this MCU**—Specifies where the workspace folder will be created. Or, click **Browse** and navigate to the desired location.
   - **...workspace name**—The name of the folder and the name of the MCU workspace file inside the workspace folder.
5. Click **Next** to display step 2 of the **MCU Wizard**. This is where you specify the project information.
6. Complete the following as desired:
   - **Project type**—Choose one of: Standard - this project type contains source code files that needs to be built in order to create a machine code (intel hex) file to load into the MCU at the start of simulation; Use External Hex File - this project type contains no files and does not need a build step.
   - **Programming language**—Active for standard **Project type** only. Select C or Assembly.
   - **Assembler/compiler tool**—Active for standard **Project type** only.
   - **Project name**—The name of the project file and the project folder that will be created inside the MCU workspace folder.
7. Click **Next** to display step 3 of the **MCU Wizard**. This is where you specify the source file that you want to create for the project.
8. Complete the following as desired:
   - **Create empty project**—Select if you do not wish to create a source file.
   - **Add source file**—Enable if you wish to create a source file.
   - **Enter the name of the source file**—Becomes active when **Add source file** is enabled. Can be an assembly or C file.
9. Click **Finish** to close the **MCU Wizard** and create the MCU related files for the MCU component.

Note If you select **Cancel** at any time, no MCU workspace is created and the MCU component is removed from the circuit schematic.

Adding and Removing Projects, Folders and Files

After you place the MCU component and create the MCU workspace, project and file, you can add or remove any projects and/or MCU files to the MCU design.

Complete the following to manipulate items in the MCU design:

1. Right-click on an MCU item in the **Design Toolbox**. The contents of the pop-up menu that appears vary depending on the MCU item type that you clicked, as detailed below:

MCU Item Type	Right-click Context Menu Items
MCU workspace	Add MCU Project…
MCU project	Add New MCU Source File… Add Existing MCU Source File… Add MCU Source Folder… Remove MCU Project Set Active MCU Project
MCU source file	Remove MCU Source File
MCU source folder	Add MCU Source File… Add Existing MCU Source File… Add MCU Source Folder… Remove MCU Source Folder

2. Select the desired menu item:
   - **Add MCU Project**—Displays a dialog box where you specify the **Project Type** that you want to create (standard/load external hex file) and the **Project Name**. The new project is created in the MCU Workspace folder of the MCU workspace that you right-clicked in the **Design Toolbox**.
   - **Add New MCU Source File**—Displays a dialog box where you select the **File type** (assembly file (`.asm`), assembly include file (`.inc`), C source file (`.c`), or C header file (`.h`)), and enter the **Filename**. The file is created in the MCU project folder or MCU source folder that you right-clicked in the **Design Toolbox**.
   - **Add Existing MCU Source File**—Displays a file browser where you select an existing source file. The selected file is copied to the MCU project folder or source folder that was selected in the **Design Toolbox**.

- **Add MCU Source Folder**—Adds an MCU folder to the MCU project folder or source folder that you right-clicked in the **Design Toolbox**.
- **Remove MCU Project**—Removes the selected MCU project from the MCU workspace.
- **Remove MCU Source File**—Removes the selected source file (`*.asm`, `*.inc`, `*.c`, `*.h`) from the MCU design.
- **Remove MCU Source Folder**—Removes the selected folder from the MCU design.
- **Set Active MCU Project**—Sets the selected MCU Project to be the active project. The active project in the **Design Toolbox** for an MCU workspace, has a blue colored square in the icon next to it. Any build operation applied to an MCU instance uses this project's files. The results of the build on the active project are also used for simulating the circuit.

Note Each file manipulation command that appears in the right-click context menus also appears in the **MCU Code Manager**. Refer to the *MCU Code Manager Overview* section for more information.

Multisim MCU Code Manager

This section describes the **MCU Code Manager**.

MCU Code Manager Overview

The **MCU Code Manager** dialog box lets you manage the MCU files in an MCU workspace and enter the build settings for each MCU project. Each MCU project has a number of build settings that consist of assembler/compiler tool information, the command line options to be used with the selected tool, and additional information on the intermediate/output files to generate during a build, and where the tools/files are located.

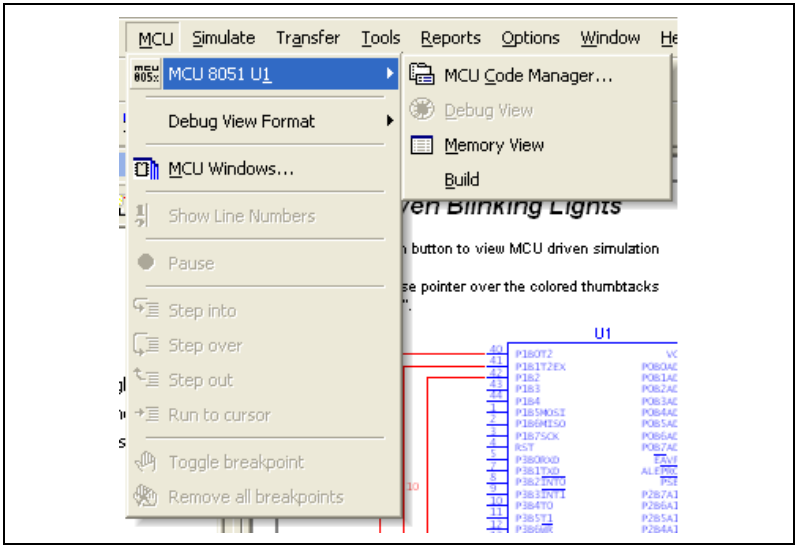
Complete the following to access the **MCU Code Manager** for a specific MCU:

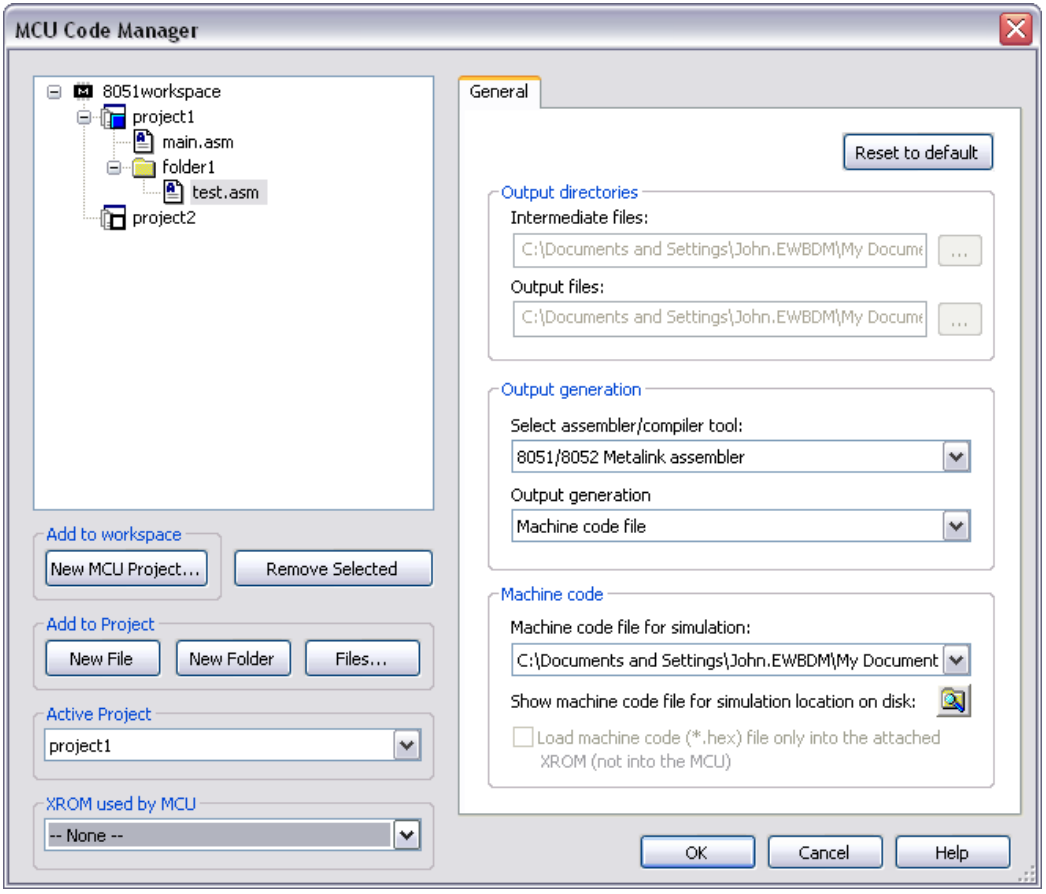
1. Select **MCU»<MCU name, e.g., MCU 8051 U1>»MCU Code Manager**.

   Or

   Right-click on the MCU workspace in the **Design Toolbox** and select **MCU Code Manager** from the pop-up that displays.

Note The **MCU** menu displays all instances of MCUs in a circuit design. In the example below, there is only one MCU "U1".





The above shows the contents of the **MCU Code Manager** for MCU U1 from the previous screen capture. `Project1` is an 8051 Metalink assembler project that has only one tab (**General**) in its project build settings.

Adding and Removing Projects, Folders and Files in the MCU Code Manager

After you place the MCU component and create the MCU workspace, project and file, you can add or remove any projects or MCU files in the MCU design.

The buttons in the **MCU Code Manager** located below the tree display of the MCU workspace correspond to the file manipulation commands that display in the right-click context menus of MCU items in the **Design Toolbox**.

Complete the following to manipulate items in the MCU design from the **MCU Code Manager**:

1. Click the **New MCU Project**, **New File**, **New Folder** or **Files** button to display the same dialog boxes as the right-click context menu commands **Add MCU Project**, **Add New MCU Source File**, **Add MCU Source Folder**, and **Add Existing MCU Source File**. Refer to the *Adding and Removing Projects, Folders and Files* section for more information.

To remove an item from the MCU workspace tree view, highlight the desired item in the tree and click **Remove Selected**.

Note You cannot remove the MCU workspace.

To select a different active project, select another project from the **Active Project** drop-down list.

To select the external ROM into which to load the MCU's machine code, select the desired **ROM** from the **XROM used by MCU** drop-down list.

Note There are no XROMs in the example in the *MCU Code Manager Overview* section, but if there was and an XROM part was selected, the **Load machine code (*.hex) file only…** checkbox would be enabled to let you control how to load the machine code into the XROM for simulation.

MCU Project Build Settings

Each MCU project in an MCU design has its own build settings that can be different from other projects inside the same workspace. The build settings available are different for each assembler or compiler tool that is supported by Multisim. The tools supported by Multisim are the 8051 Metalink Assembler, the Microchip PIC Assembler, the Hi-Tech 8051 C Lite compiler and the Hi-Tech PICC Lite compiler.

To configure the build settings, use the tabs that appear on the right of the **MCU Code Manager** dialog box. The tabs that appear, and their content, correspond to the project that is currently selected in the MCU workspace tree on the left.

The following table lists the assembler/compiler families of tools that are supported by Multisim and the **MCU Code Manager** tabs that are available for configuring those tools:

Assembler/compiler Family	MCU Code Manager Tabs
Metalink - 8051/8052 Cross Assembler	General
Microchip (PIC) - MPASM absolute and relocatable assembler - MPLINK linker - MPLIB librarian	General C/Assembly Link Library
Hi-Tech - 8051 C Lite compiler - HLINK linker - LIBR librarian	General C/Assembly Library
Hi-Tech - PICC Lite compiler - LINK linker - LIBR librarian	General C/Assembly Library

Documentation for Supported Tools

User documentation for the supported tools is found at the following:

- **Metalink assembler**—`...\Program Files\National Instruments\Circuit Design Suite 10.1\documentation\MetaLink 8051 Cross Assembler Guide.pdf`.
- **Microchip MPASM assembler**—`...\Program Files\National Instruments\Circuit Design Suite 10.1\documentation\DS-33014J.pdf`.
- **HI-TECH 8051 C compiler**—`...\Program Files\HI-TECH Software\HC51\lite\9.60\docs\manual.pdf`.
- **HI-TECH PICC Lite compiler**—Go to `www.htsoft.com` and download HI-TECH PICC (V9.50PL2).

8051 Workspace Build Settings Example

In the example shown in the *MCU Code Manager Overview* section, `project1` is selected. The Metalink assembler is selected for the project. Since this is an absolute assembler, the build settings for it are very simple. Consequently, only the **General** tab is available to configure the build settings. The **Intermediate files** and **Output files** paths are disabled because the Metalink assembler does not allow you to redirect your output.

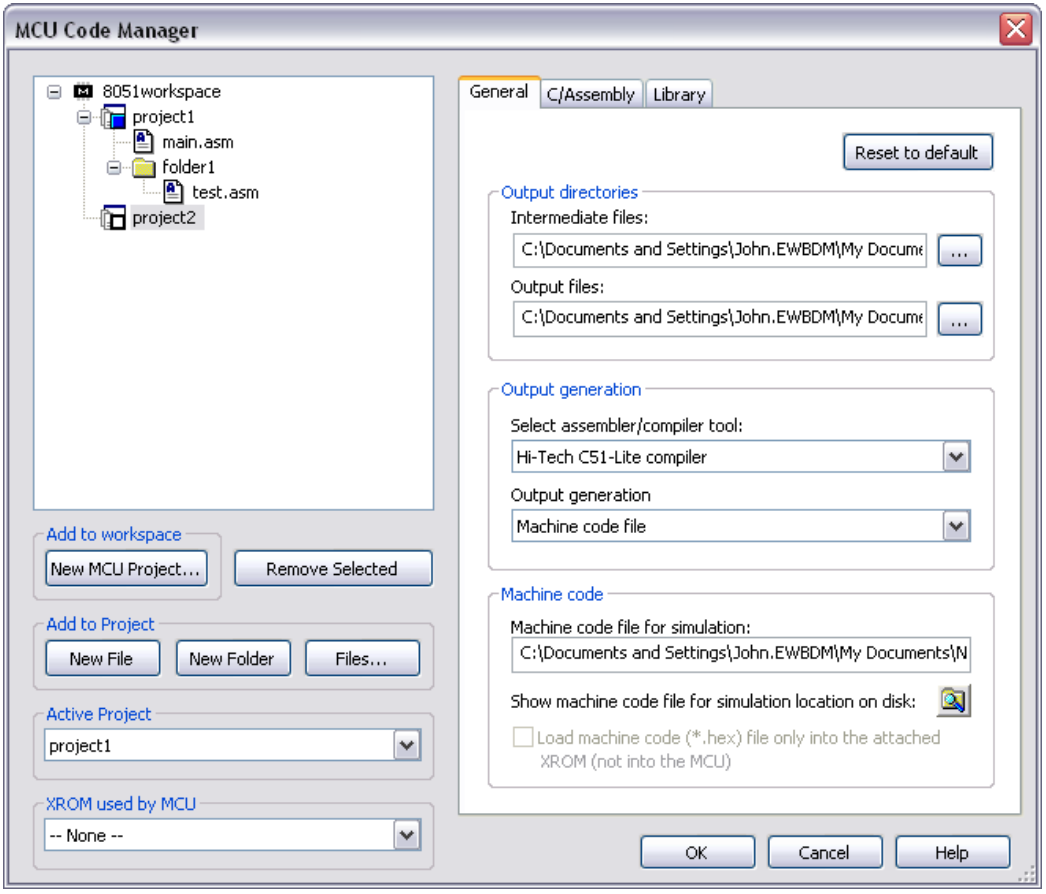
After the output machine code file (`*.hex`) is generated, the contents must be loaded into the Multisim simulation engine in order to simulate the circuit.

To the set the output `*.hex` file generated during a build step, select the file from the **Machine code file for simulation** drop-down list.

Complete the following to see the whole path name of output files:

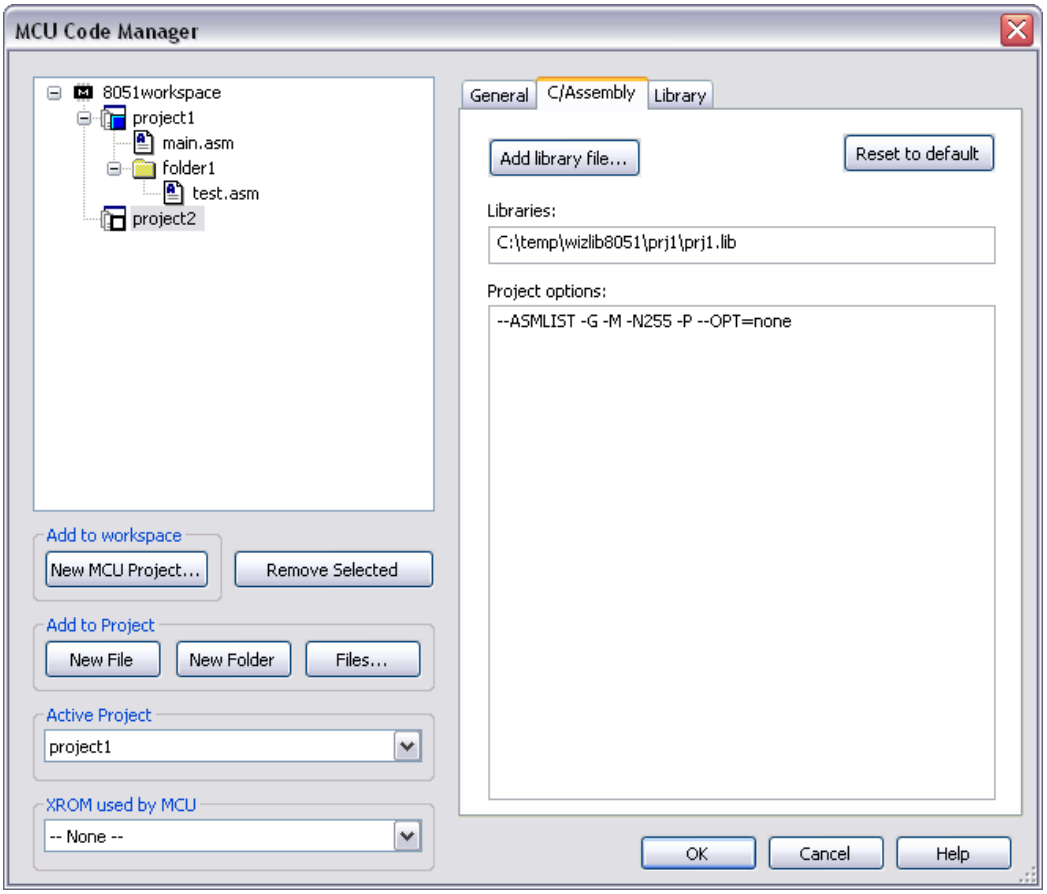
1. Click **Show machine code for simulation location on disk**. Windows Explorer opens in the folder where the output file will be located.

In the screen capture below, `project2` of `8051workspace` is selected in the **MCU Code Manager**. `Project2` has the "Hi-tech C51-Lite compiler" selected as the tool. This is a C compiler which compiles and links code in order to produce an output machine code file (`*.hex`), and is more advanced than the Metalink assembler. It has three tabs (**General**, **C/Assembly**, and **Library**) to configure its build settings.



The assembler or compiler type determines the type of output target files that you can generate. In the screen capture above, the Hi-Tech C51 Lite compiler is chosen. Consequently, there are two types of target files that you can create: a machine code file and a library file. If you choose a library file, the compiler generates object files from the source files and places them in the specified library file instead of creating a machine code file. The library file is configured in the **Library** tab.

There is a **C/Assembly** tab (see below) in addition to the **Library** tab available to allow you to configure the Hi-Tech tools further. One of the advanced features of the Hi-Tech tools is that intermediate and output files generated by the Hi-Tech C51 Lite can be redirected - therefore the **Intermediate files** and **Output files** paths are enabled in the **General** tab.

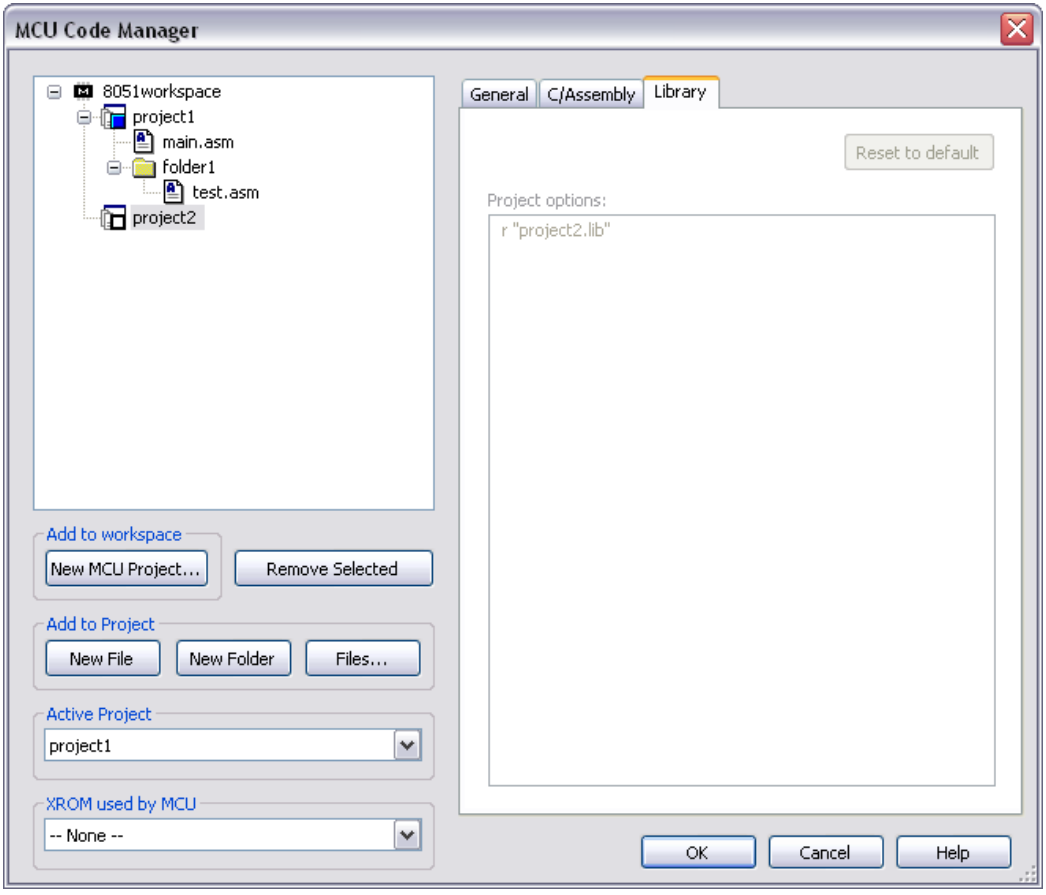


The above shows the **MCU Code Manager** for MCU U1 in the *MCU Code Manager Overview* section. The **C/Assembly** tab of the Hi-Tech C51 Lite compiler for `project2` is displayed.

The **Project options** edit box contains the compiler line options for the C51 Lite compiler. The default settings are normally sufficient to build an 8051 Hi-Tech C program but you can customize the options based on the Hi-Tech C51 Lite documentation. Refer to the *Documentation for Supported Tools* section for more information.

The **Libraries** edit box is where you can specify pre-existing libraries to be linked into the final build. In the above screen capture, one library file will be linked into the final build. Use the **Add library file** button to display a file browse window to find the library that should be linked into the build and automatically add the library file path to the **Libraries** edit box.

The **Library** tab, as shown below, is enabled only if the output target selected in the **General** tab is a library. The **Project options** edit box in the **Library** tab displays the command line options used when the Hi-Tech tools generate the library. The figure below shows that the **Library** options are disabled since the output selected for `project2` is a hex file. The Hi-Tech compiler documentation includes details on it command line options. Refer to the *Documentation for Supported Tools* section for more information.

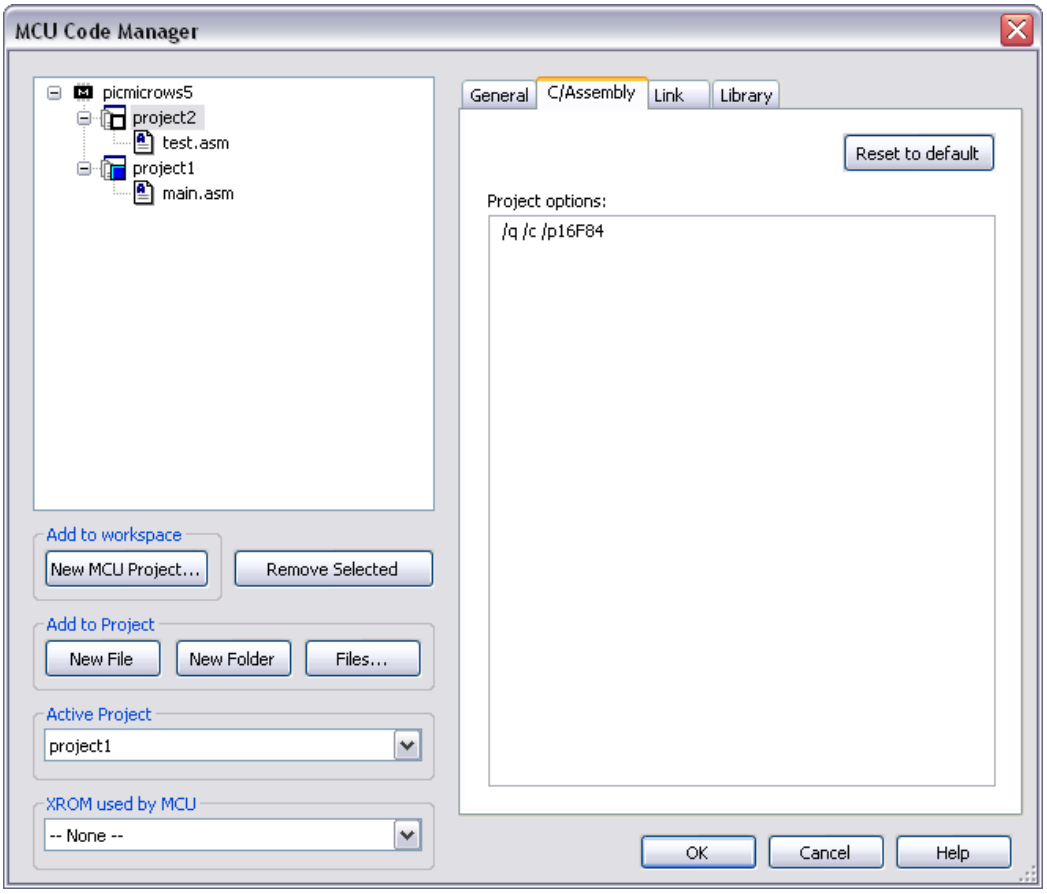


The above figure shows the **MCU Code Manager** for MCU U1 in the *MCU Code Manager Overview* section. The **Library** tab of Hi-Tech C51 Lite compiler for `project2` is displayed.

The tabs for the other assembler/compiler files are all similar except that the command line options in the **C/Assembly** and **Library** tabs are different for each tool. Refer to the assembler/compiler documentation for each tool for more information on the options that can be used with them. Refer to the *Documentation for Supported Tools* section for more information.

The tab that was not shown in the above example is the **Link** tab. The Hi-Tech compilers have the ability to pass on the command line options to their linker tool from the compiler. This means that the command line options for both the compiler and the linker can be specified all at once in the command line options for the compiler, which is when no **Link** tab is required.

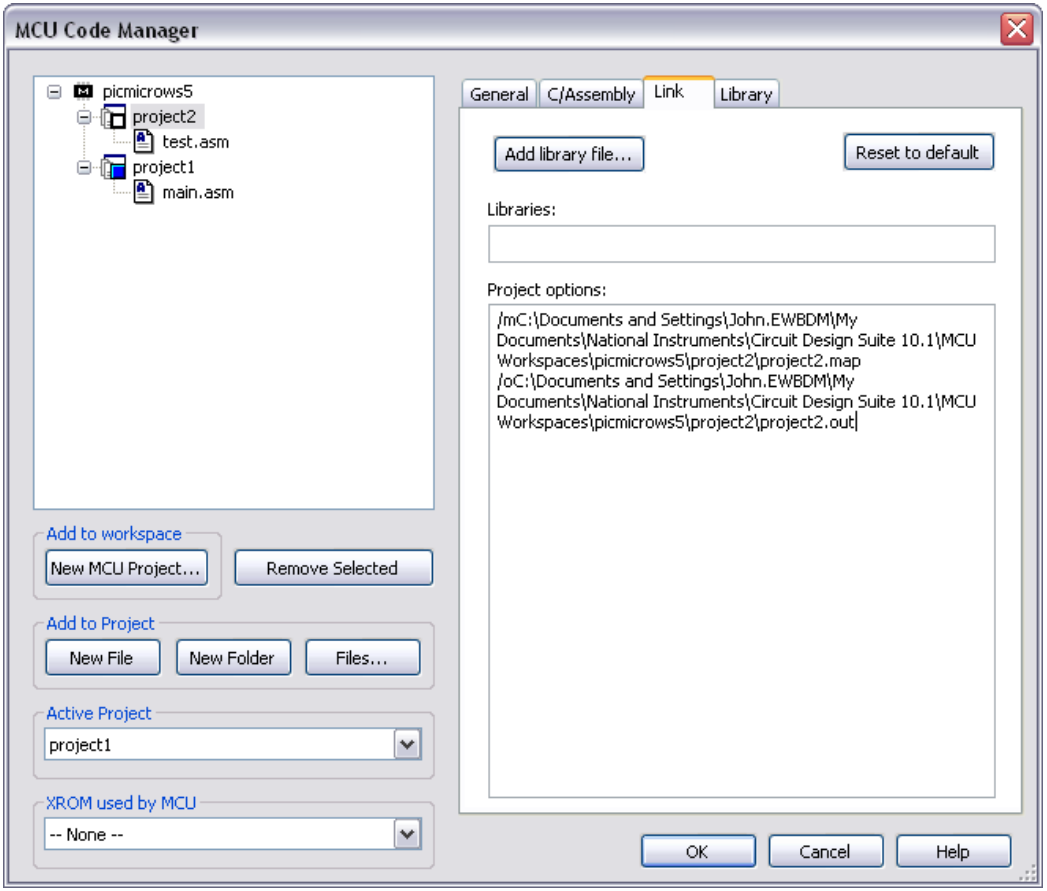
The Microchip relocatable assembler requires the **Link** tab to configure the command line options for MPLINK, its linker. The **C/Assembly** tab for the Microchip tools displays the options for the compiler, MPASM. The figure below shows an example of the **C/Assembly** tab for the Microchip assembler.



Use the **Link** tab for the Microchip assembler to add any libraries that need to be linked into the final build and the linker options that have to be passed to MPLINK. The next figure shows an example of the **Link** tab.

Note The Microchip assembler has both absolute and relocatable assembler. To use MPASM in relocatable assembler mode, you must add the `/o` option to the **Project options** edit box in the **C/Assembly** tab shown above, to generate the object files (`*.o`) used in the linking stage. Otherwise the object files do not generate and only the machine code file

(`*.hex`) will be generated as the output file. You must also write your code in a relocatable fashion using the `CODE` directive before the code can be assembled in relocatable mode.



The **Link** tab for the Microchip MPLINK tool displays command line options in the **Project options** edit box and provides a way of specifying the libraries that should be linked into the final build.

Loading an External Hex File Project

The build settings described in the previous section are for standard projects that have an assembler/compiler tool associated with them. The second type of project, (load external file project) requires no tools. Only the machine code file path must be configured.

During simulation, the **Machine code file for simulation (*.hex)** specified in the edit box containing the opcodes is loaded into the simulation and run. It is also possible to load the code into an XROM part associated with the MCU as is done for standard projects.

Building an MCU Workspace

To build an MCU workspace, the proper build settings must be configured in the **MCU Code Manager** as described in the previous sections and all necessary source files must be present in the active MCU project. The assembler/compiler tools selected in the **General** tab of the **MCU Code Manager** for the active project, are invoked during the build to generate the appropriate files. These files can be machine code files (`*.hex`), object files, libraries, listing files and any other intermediate or output files described in the tool's documentation. Refer to the *Documentation for Supported Tools* section for more information.

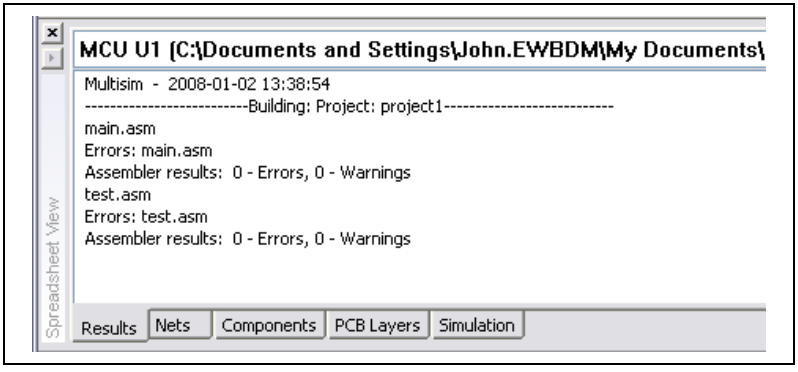
Complete the following to build the MCU Workspace:

1. From the MCU menu, navigate to the submenu of the MCU that you wish to build and select **Build**.

   Or

   Right-click on the workspace or active project item in the **Design Toolbox** to display the context menu and select **Build**.

The results of the build are displayed in the **Results** tab of the **Spreadsheet View**. An example is shown below. If the build is successful, you will have either a `*.hex` or `*.lib` file in the output path where your output files should go. You can verify this by using Windows Explorer to view the files in that subdirectory.



Errors and Warnings

If there are errors or warnings in your build, they will be displayed in the **Results** tab.

Complete the following to find an error:

1. Double-click on the line displaying the error in the **Results** tab. The file that contains the error comes into focus with the cursor placed on the line corresponding to the error.
2. Correct the errors in the source file and then build the MCU workspace again, continuing in the same manner until the build is error-free.

Simulation of Machine Code File

Standard Projects

After building the MCU workspace successfully, simulate the circuit containing the MCU component by selecting **Simulate»Run**. If you try to run the simulation without building the MCU workspace first, Multisim prompts you to build the MCU.

When you build the MCU workspace, listing files (`*.lst`) are generated that are loaded into Multisim for debugging purposes. Depending on the assembler/compiler tool, the listing files contain a combination of data such as opcodes, where the opcodes should reside in the MCU internal ROM, error information, assembly instructions corresponding to the opcodes and the matching line of source code for each assembly instruction. If the listing file is missing, or cannot be loaded into Multisim, then no instruction level debugging can be performed on the MCU.

The intel hex format machine code file (`*.hex`) that is generated by the build is also loaded into Multisim at the start of simulation so that the MCU component's internal ROM representation can be loaded with the opcodes that should be executed during simulation. This machine code file path is specified in the **General** tab of the **MCU Code Manager**. You can check that the correct machine code file exists in the machine code file path if Multisim has trouble loading it.

Load external hex file projects

Standard MCU projects generate the `*.hex` file during the build step. However the `*.hex` file does not have to be generated for load external hex file projects. The machine code file specified in the **General** tab of the **MCU Code Manager** is loaded at the start of simulation. It is still possible to debug through disassembled assembly instructions for the MCU during simulation in the **Debug view** since the opcodes are disassembled when the `*.hex` file is loaded.

Simulating MCU with XROM

The **General** tab of the **MCU Code Manager** allows an XROM (external ROM) component to be used with the MCU. There are three scenarios that can happen relating to XROM:

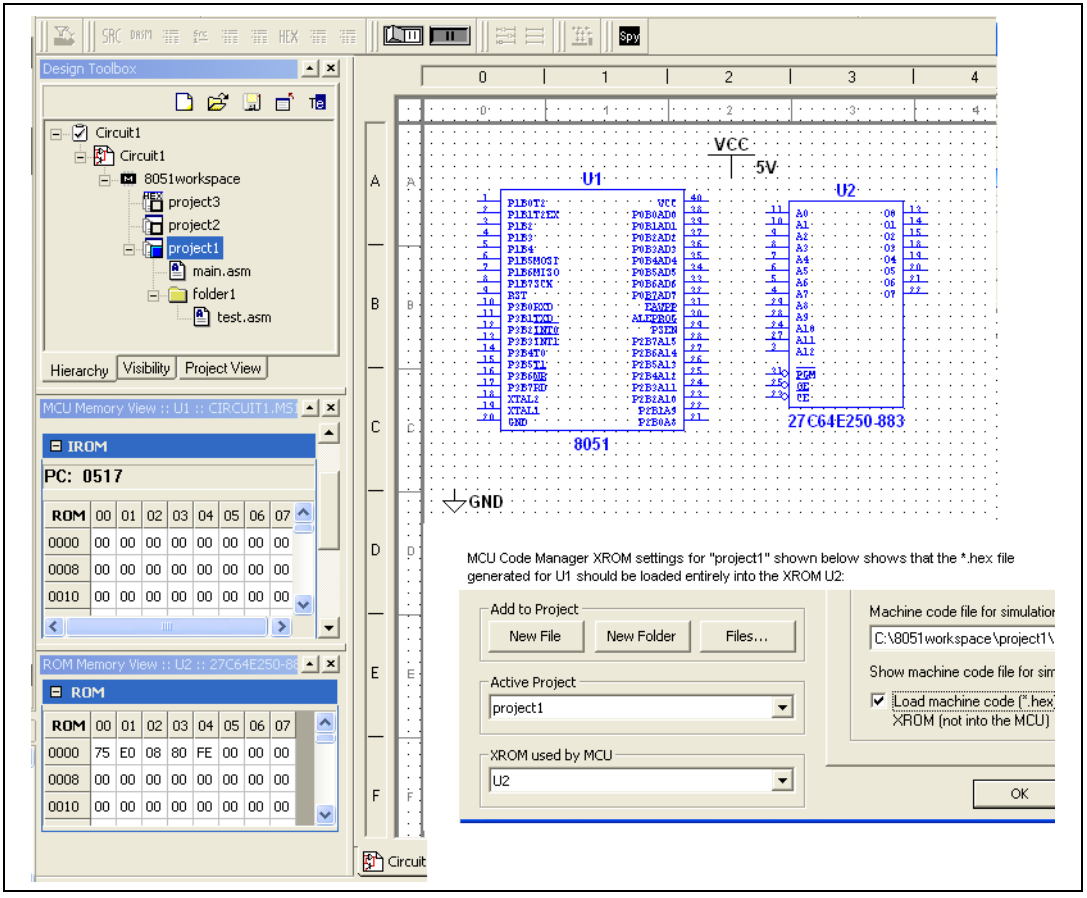
- If no XROM is selected, the machine code (`*.hex`) file opcodes are loaded entirely into the MCU as described in all of the other examples above for standard and load external hex file projects.
- If an XROM is selected and the **Load machine code (*.hex) file into XROM** checkbox is selected, the machine code file generated by the MCU build is loaded entirely into the XROM. None of the opcodes are loaded into the MCU; the internal ROM of the MCU will be empty (filled with no-ops).
- If an XROM is selected and the **Load machine code into XROM** checkbox is not selected, the machine code file generated by the MCU build is loaded into the MCU's internal ROM, but any opcodes that should be loaded into memory address extending beyond the physical size of the internal ROM are loaded into the XROM at those very high memory addresses.

The example below shows a circuit containing an MCU (U1) and an XROM (U2) where the opcodes in the `*.hex` file generated during the build step for a standard project `project1` is loaded entirely into the XROM (U2). The simulation is paused and the **XROM Memory View** shows that the opcodes are indeed loaded into U2 and that there is nothing in U1 as shown in the **MCU Memory View**. The bottom right side is a snippet from the **MCU Code Manager** settings for `project1` that shows the XROM settings. The `*.hex` file is generated using the 8051 Metalink assembler from the `main.asm` source code:

```
$MOD51    ;This includes 8051 definitions for the
metalink assembler
```

```
        MOV ACC,#08H
LOOP:
        JMP LOOP

        END
```



The circuit containing MCU U1 and XROM U2 is configured so that during simulation, the opcodes generated from the source code for `project1` are loaded entirely into the XROM as shown in the **XROM Memory View**.

Multisim MCU Source Code Editor

This section describes the MCU **Source Code Editor**.

Opening a Source Code File

Complete the following to open a source code file in the editor:

1. Double-click on the name of the source code file (for example, `main.c` or `main.asm`) in the **Design Toolbox**.
2. Optionally, select **MCU»Show Line Numbers** if you wish to display line numbers in the source code view.

Building Source Code Files

Source code files must be built to create a machine code (intel hex) file to load into the MCU at the start of simulation. Also, any changes in source code files will not take effect until the MCU project is built.

Complete the following to build source code files:

1. Select **MCU»<MCU name, e.g., MCU 8051 U1>»Build**.

   Or

   Right-click on the desired source code file in the **Design Toolbox** and select **Build**.

The **Build** command invokes the appropriate compilers, assemblers and linkers, and then loads the error, listing and hex files that they generate into Multisim. Errors, warnings and messages display in the **Results** tab of the **Spreadsheet View**.

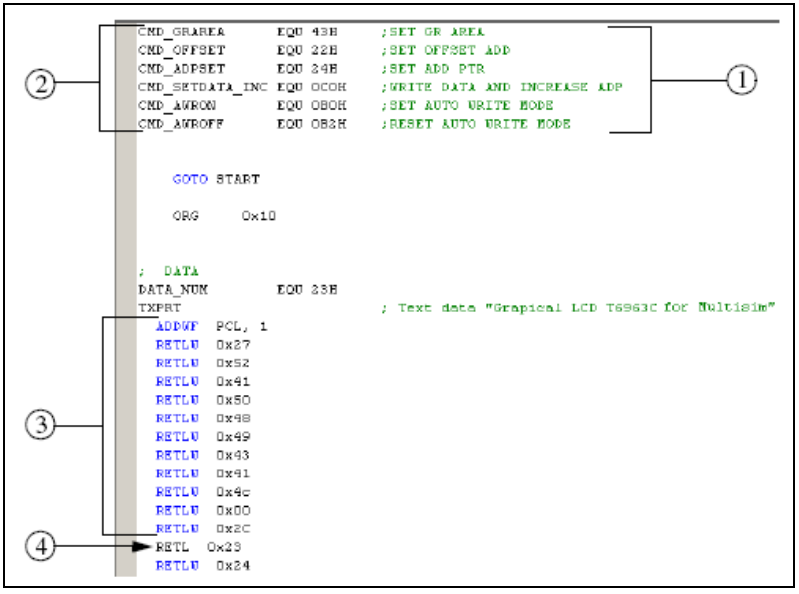
If the build is successful, the hex file is loaded into the appropriate MCU or external ROM. If it is unsuccessful, the ROM is filled with zeros.

To go to the line of source code associated with an error, double-click on the error in the **Results** tab.

Source Code View

Sections of the MCU **Source Code Editor** are color-coded to denote the text type:

- Blue text—Key words (3).
- Green text—Notes (1).
- Black text—All other text, including data (2), errors (4).



1	Notes	3	Key Words
2	Data	4	Error

Note If you type text with an error (spelling, incorrect syntax) into a key word section of the code, the text changes color from blue to black.

To save the source code file, select **File»Save** from the source code view.

To print the source code file, select **File»Print** from the source code view.

Complete the following to find a text string in the source code:

1. Select **Edit»Find** to display the **Find** dialog box.
2. Enter the desired string and click **Find Next**.

MCU Debugging Features

This section describes the Multisim MCU Debugging Features.

You should also read the Multisim MCU Tutorial chapter of the *Getting Started with NI Circuit Design Suite* manual.

Definitions

After an MCU project has been compiled/assembled into a hex file and loaded by Multisim, several different pieces of information are available:

- **source code**—The original text that was entered by you and passed to the compilers.
- **listing assembly**—The assembly text that is output by the compiler into the listing files. This form assembly has variable names and other extra notations from the compiler which is useful when debugging or analyzing the output of a C program. If the original source code was assembly the listing assembly is often identical; only when macros are used does it differ.
- **disassembly**—Created by Multisim from the raw hex opcodes. This is often similar to the listing assembly, however, it usually contains absolute values instead of variable names, except for the jump/goto instructions which use the named labels.

Multisim's MCU functionality includes code and debugging windows in addition to Multisim's standard circuit windows. Inside the debugging windows a dropdown list allows you to pick different debugging views. The views show different subsets of the same information, and the debug window settings let you further filter and customize what the information that is presented in these views.

Opening a Debug View

After an MCU's project has been built (without errors) using the **Build** command the debug window becomes accessible.

To open the debug window for a particular MCU, select **MCU»<MCU name, e.g., MCU 8051 U1>»Debug View**.

This opens a new window in Multisim with a drop-down list at the top that you use to select between the different views.

There is one view available for each source code file, as well as one for the disassembly of the entire project. By default, when you open a **Debug View**

the project disassembly view which is a complete disassembly of the ROM for the relevant MCU opens. In the project disassembly view, every memory address displays from address zero to the maximum address of the ROM. The project disassembly view always shows the disassembly (not the listing assembly) because it shows memory addresses which may not have a corresponding listing (such as the opcodes from statically linked libraries).

To switch to another of the available views, left-click on the drop-down list.

The source file debug listing view is similar to the listing text that a compiler generates when it builds a single file. It contains a subset of the information available in the complete ROM disassembly, but the information is sorted into the order of the original source code, rather than the memory address.

Note The debug file view can also be launched directly—right-click the appropriate file in the **Design Toolbox** and select **Debug View**.

Debug Window Settings

Complete the following to select the information to view in the **Debug view**:

1. Select **MCU»Debug View Format** and select the desired option.

   Or

   Right-click on the **Debug view** and enable the desired option from the **Debug View Format** menu item.

   Or

   Enable the desired button in the **MCU** toolbar.

   These settings are saved as part of each MCU component, thus if you have two components they can have different settings, however all the views inside the debugging window will have the same settings.

MCU Toolbar Buttons

Button	Description
	Source Code button. Enable to debug through the original source code or the assembly from the compilers listing files/multisim disassembler. Applies to the file debugging views, not to the project disassembly.
	Disassembly button. Determines whether you see the listing assembly text or the version from the disassembler. Applies to the file debugging views, not to the project disassembly.
	Show Secondary Language as Comments button. Enables/disables one of two things. If you are looking at the project disassembly view or if you are looking at the listing assembly or disassembly of a file it will add the original source code corresponding to those lines as comments above or beside the assembly. If you are viewing the source code debug listing then this will show the corresponding listing assembly/disassembly as comments below the source code.
	Show Line Numbers button. Displays the line numbers in the **Debug View**.
	Show Memory Addresses in Debug View button. Enables/disables the showing of memory addresses for any text that shows source code in any of the debug listing views.
	Show Memory Addresses in Assembly Code button. Enables/disables the showing of memory addresses for any text that shows listing assembly or disassembly in any of the debug listing views.
	Show **Hex Opcodes in Assembly Code** button. Enables/disables the showing of the hex values of opcodes for any text that shows listing assembly or disassembly in any of the debug listing views.

Button	Description
	Show Jump/Goto Labels button. Enables/disables the showing of the jump/goto labels for any text that shows listing assembly or disassembly in any of the debug listing views.
	Show Headings Above Code button. Enables/disables the showing of the heading comments above each block of source or assembly code which labels the different fields being shown in the text.

Simulation Markers

The margin that runs down the left side of the MCU debug windows and **Source Code View** is where breakpoints and simulation markers are shown. When you pause the simulation, a yellow arrow in the bar on the left side of the source code indicates the current line of execution which corresponds to the current value of the program counter (PC) in that MCU. Since each line of source code may actually refer to many opcodes, the current program counter may actually be located somewhere in the middle of the opcodes for a line when you pause the simulation.

The simulation marker also appears in the debug view, which is where most of the debugging is done. If your debug view is currently set to show source code, it behaves the same way the source code window does. If you are viewing the disassembly or the listing assembly, the indicator is located on the line that contains the memory address that corresponds to the program counter. Since the debug view corresponds to a single MCU it has a maximum of one simulation marker. However, the source code view may have more than one indicator if there are two MCU instances which share that MCU project (this can only happen if you have MCU components inside subcircuits or hierarchical blocks).

Breakpoints

You can place breakpoints in either source code or debugging views. Breakpoints can only be added to lines that correspond to real memory addresses, therefore they cannot be placed on comments or other lines that do not map to any real opcodes. Before breakpoints can be placed, you must build the project. Refer to the *Building Source Code Files* section for more information.

Complete the following to place a breakpoint:

1. Double-click on the margin at the left edge of the source code or debugging window. Breakpoints are displayed as a red dot in the left margin beside any line of source code or assembly.

   Or

   Select the line of source code or debug listing text that the breakpoint should be placed on and choose the **Toggle Breakpoint** button in the **Simulation** toolbar.

   Or

   Right-click on the line of source code or debugging text and choose **Toggle Breakpoint** from the context menu.

To remove a breakpoint, repeat any of the above, or, click the **Remove All Breakpoints** button.

A breakpoint placed on a source code window will also be visible in the debug view. These are simply two different ways of looking at the same information. Breakpoints that are placed on listing assembly or disassembly views that are associated with a line of source code, even if they are in the 'middle' of the opcodes for that source code, show up in any source code views. Breakpoints placed on source code correspond to the first memory address associated with that line of source code in the disassembly or listing assembly.

After each build the breakpoints may be adjusted based on the changes to the source code and may shift slightly in both views. This happens because breakpoints placed in the source code view will remain 'locked' to the line of source code on which they were placed even if changes result in the compiler moving the memory address of the opcodes that correspond to the source code. Breakpoints that are placed on the listing assembly or the disassembly remain 'locked' to the exact memory address, so if the source code is modified and then built they may shift their position in the source code view to the source code that now corresponds to that memory address.

If MCU components are placed in hierarchical blocks or subcircuits it becomes possible to have two different 'instances' of the same component in your circuit. In this case both MCU components share the same MCU project and source code files, and therefore a breakpoint will be placed in each of MCU components. Toggling the breakpoint on or off in the source code toggles them all on or off, however toggling them on or off in the debug view only removes the breakpoint for the corresponding component. The breakpoint marker in the source code view only disappears when all the breakpoints in all the MCU's debug windows are removed.

Stepping and Breaking

The controls for simulation stepping and breaking into an MCU are located in the **Simulation** toolbar, and also in the **MCU** menu.

When you select the **Pause Simulation** toolbar button menu item during a simulation, the simulator stops at the next SPICE timestep, possibly leaving the CPU simulator in mid-instruction. If you press the **Pause Simulation at Next MCU Instruction Boundary** button, the simulation waits slightly longer before breaking. If the currently activated window and view is displaying listing assembly or disassembly then the simulation break does not cause the simulation to stop until it reaches the begining of a new MCU opcode. If the current window and view is displaying source code then the simulator continues until it reaches the start of an opcode which is associated with the start of the next line of source code.

The stepping functions are **Step into**, **Step over**, **Step out** and **Run to cursor**. All of these buttons work by starting a simulation and then stopping when a certain point is reached. The behavior depends on whether the view that is active is showing source code (as the primary language) or listing assembly/disassembly.

If you click the **Step into** toolbar button, the simulator starts and runs until it reaches the next line of source code or listing assembly/disassembly. If the current opcode is the start of a function call, the **Step into** command follows the call to its new location and breaks at the next opcode.

If you click the **Step over** button, the simulator starts and runs until it reaches the next line of source code or listing assembly/disassembly. If the current opcode is the start of a function call then **Step over** does not follow the call, but rather breaks at the first opcode after the functions return. Another way of thinking of this is that the simulator breaks at the start of the first opcode where the call stack depth is less than or equal to the current value.

If you click the **Step out** button, the simulator starts and runs until it reaches the line of source code or listing assembly/disassembly that immediately follows the return from the current function. Another way of thinking of this is that the simulator breaks at the start of the first opcode where the call stack depth is less than the current value.

For a demonstration of these functions, refer to the Multisim MCU Tutorial chapter of *Getting Started with NI Circuit Design Suite*.

Memory View

To display an MCU's memory view, select **MCU»<MCU name, e.g., MCU 8051 U1>»Memory View**. A dockable window opens which contains several spreadsheet-like grids of data.

These grids contain data only while the simulation is running and in a paused state. When paused, they represent a snapshot of the internal registers and memory banks for that component at a point in simulation time. Different MCU's have different subsets of these grids depending on the features of that microchip. While the simulation is paused these values can be used to examine the internal state of the component and also to manually modify them.

To modify the values, click on the corresponding grid entry and type in new values in the same format as the previous values.

External RAM and ROM components also have memory views. When the simulation is paused, the you can view the contents of the memory for these parts. Refer to the *Component Reference help file* for more information.

16

Automation API

Multisim provides an Application Programming Interface (API) that allows for the automation of circuit simulation and analyses using Multisim. This is a COM interface that you can use to programmatically control Multisim without directly interacting with it.

The COM interface supports a limited set of circuit editing features. For example, from the client application, you can open a Multisim circuit, simulate the circuit, and then retrieve simulation results for further processing in the client. In addition, the client program can select output variables and substitute client-supplied data for power supplies. Other applications include:

- **Component Evaluator**—This lets you test multiple components under the same circuit topology. You can replace the device under test with others from the database and compare the results. You can also vary the values of the resistors, capacitors, and inductors in the circuit to determine their impact.
- **Stress Analysis**—This is akin to an automated worst-case analysis, where circuit elements are successively replaced by elements that are within the original elements' tolerances to see whether power consumption or other such variables exceed allowed values.
- **Fault Diagnosis**—This simulates a number of faults in a circuit (for example, short circuits, burned-out components, missed connections) that can be compared against a faulty prototype PCB to try to diagnose where the error in the prototype actually lies. This may require the addition of extra circuit elements to simulate the faults.

The COM interface in Multisim can be used by clients written in any COM-aware programming language, such as NI LabVIEW, Visual Basic or C++.

Refer to the *Multisim help file* for complete information on the Automation API.

A

Menus and Commands

The following sections contain brief descriptions for the commands in the Multisim menus.

Some of the described menu items may not be available in your edition of Multisim. Refer to the NI Circuit Design Suite Release Notes for a list of the features in your edition.

File Menu

The following table describes the commands found in the **File** menu.

Command	Description
New»Schematic Capture	Opens an untitled circuit window that can be used to create a circuit. The new window opens using your default circuit preferences. Until you save, the circuit window is named “Circuit#”, where “#” is a consecutive number. For example, “Circuit1”, “Circuit2”, “Circuit3”, and so on.
Open	Opens a previously created circuit file or netlist. Displays a file browser. If necessary, change to the location of the file you want to open. You can also open Ulticap and Orcad files. Refer to the *Importing Files with Other Formats* section of Chapter 13, *Transfer/Communication*, for more information. You can open files created with Version 5 of Electronics Workbench, files created in previous versions of Multisim, and netlist files.
Open Samples	Opens a folder containing sample and getting started files.
Close	Closes the active circuit file. If any changes were made since the last save of the file, you are prompted to save those changes before closing.
Close All	Closes all circuit files. If any changes were made since the last save of the file, you are prompted to save those changes before closing.

Command	Description
Save	Saves the active circuit file. If this is the first time the file is being saved, displays a file browser. Optionally, change to the desired location for saving the file. You can save a circuit file with a name of any length. The extension `.ms10` is added to the file name automatically. For example, a circuit named `Mycircuit` is saved as `Mycircuit.ms10`. To preserve the original circuit without changes, choose **File» Save As**.
Save As	Saves the current circuit with a new file name. The original circuit remains unchanged. Use this command to experiment safely on a copy of a circuit, without changing the original.
Save All	Saves all open circuit files.
New Project	Creates a new project for grouping together related circuit designs. Refer to the *Setting up Projects* section of Chapter 4, *Working with Larger Designs*, for more information.
Open Project	Opens an existing project. Refer to the *Working with Projects* section of Chapter 4, *Working with Larger Designs*, for more information.
Save Project	Saves the current project. Refer to the *Working with Projects* section of Chapter 4, *Working with Larger Designs*, for more information.
Close Project	Closes an open project. Refer to the *Working with Projects* section of Chapter 4, *Working with Larger Designs*, for more information.
Version Control	Backs up or restores a project. Refer to the *Version Control* section of Chapter 4, *Working with Larger Designs*, for more information.

Command	Description
Print	Opens a standard **Print** dialog box. Refer to the *Printing the Circuit* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Print Preview	Previews the circuit as it will be printed. Opens a separate window, where you can move from page to page and zoom in for details. You can also print what you preview. Refer to the *Printing the Circuit* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Print Options»Print Circuit Setup	Allows you to set the page margins, page orientation (landscape or portrait), zoom level and output options. These settings apply only to the current circuit. Refer to the *Printing the Circuit* section of Chapter 2, *Schematic Capture - Basics*, for more information about these fields. If you change the page orientation using the **Page Circuit Setup** dialog box, the change is also reflected in the **Print Setup** dialog box.
Print Options»Print Instruments	Prints the faces of any or all of the instruments in your circuit. Refer to the *Printing Instruments* section of Chapter 9, *Instruments*, for more information.
Recent Designs	Displays a list of the eight most recently saved circuit files. To re-open a file, select it from the list.
Recent Projects	Displays a list of the eight most recently saved projects. To re-open a project, select it from the list.
Exit	Closes all open circuit windows and exits Multisim. If you have unsaved changes in any circuits, you are prompted to save or cancel them.

Edit Menu

The following table describes the commands found in the **Edit** menu.

Command	Description
Undo	Undoes the most recently performed action.
Redo	Redoes the most recently performed undo.
Cut	Removes selected components, circuits or text. What you cut is placed on the Windows clipboard so you can paste it elsewhere.
Copy	Copies selected components, circuits or text. The copy is placed on the Windows clipboard. You can then use the **Paste** command to paste the copy elsewhere, including other applications such as word processors.
Paste	Places the contents of the clipboard. The cursor shows a "ghosted" image of the item to be pasted. Click to indicate where the item is to be pasted.
Delete	Permanently removes selected components or text. Does not place the selection on the clipboard and does not affect anything currently on the clipboard. Use the **Delete** command with care. Deleted information cannot be retrieved, although a delete can be recovered using the **Undo** command.
Select All	Selects all items in the active circuit window. To select all but a few items, use the **Select All** command and then deselect the ones you don't want using `<Ctrl>`.
Delete Multi-Page	Displays the **Delete Multi-Page** dialog box where you select the page to be removed from a multi-page circuit file. Refer to the *Delete Multi-Page Dialog Box* section of Chapter 4, *Working with Larger Designs*, for more information. Use the **Delete Multi-Page** command with care. Deleted information cannot be retrieved, although a delete can be recovered using the **Undo** command.
Paste as Subcircuit	Pastes the selection on the clipboard as a subcircuit. Refer to the *Hierarchical Design* section of Chapter 4, *Working with Larger Designs*, for more information.

Command	Description
Find	Displays the **Find Component** dialog box. Refer to the *Finding Components in Your Circuit* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Graphic Annotation»Pen Color	Displays a palette where you can change the color of lines for graphic elements, excluding placed text.
Graphic Annotation»Pen Style	Displays options for the appearance of lines for graphics elements, excluding placed text.
Graphic Annotation»Fill Color	Displays a palette where you can change the fill color of graphic elements like rectangles.
Graphic Annotation»Fill Type	Displays options for the appearance of the fill of graphic elements like rectangles.
Graphic Annotation»Arrow	Displays available arrows used to end lines and other graphic elements.
Order	Either sends a selected graphic item to the back, or brings it to the front.
Assign to Layer	Assigns the selected item to an annotation layer. Refer to the *Visibility Tab* section of Chapter 1, *User Interface*, for more information.
Layer Settings	Displays the **Visibility** dialog box. The settings found in this dialog box are also found in the **Visibility** tab of the **Sheet Preferences** dialog box. Refer to the *Sheet Properties - Visibility Tab* section of Chapter 1, *User Interface*, for more information.
Orientation»Flip Horizontal	Flips the selection horizontally. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Orientation»Flip Vertical	Flips the selection vertically. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Orientation»90 Clockwise	Rotates the selection 90 degrees clockwise. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Orientation»90 CounterCW	Rotates the selection 90 degrees counter-clockwise. Refer to the *Rotating/Flipping Placed Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Command	Description
Title Block Position»Bottom Left	Places the title block at the bottom left of the sheet.
Title Block Position»Bottom Right	Places the title block at the bottom right of the sheet.
Title Block Position»Top Left	Places the title block at the top left of the sheet.
Title Block Position»Top Right	Places the title block at the top right of the sheet.
Symbol»Title Block	Does an in-place edit of a selected component's symbol, or a selected title block.
Font	Displays the **Font** dialog box, where you can change various font properties. Contains the same information as the **Font** tab in the **Sheet Properties** dialog box. Refer to the *Sheet Properties - Font Tab* section of Chapter 1, *User Interface*, for more information.
Comment	Edits the contents of a selected comment. Refer to the *Adding a Comment* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Forms/Questions	Displays the **Edit Form** dialog box where you enter and edit questions related to the circuit. Refer to the *Linking a Form to a Circuit* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Properties	Opens the properties dialog box of the item selected in the active window. If there is no item selected, the **Sheet Properties** dialog box displays.

View Menu

The following table describes the commands found in the **View** menu.

Command	Description
Full Screen	Shows the workspace without any toolbars or menu items showing.
Parent Sheet	Displays the active subcircuit or hierarchical block's parent sheet. Refer to the *Viewing a Parent Sheet* section of Chapter 4, *Working with Larger Designs*, for more information.

Command	Description
Zoom In	Increases the magnification of your view of the circuit.
Zoom Out	Decreases the magnification of your view of the circuit.
Zoom Area	Zooms in on selected components when the workspace is displayed at less than 100%.
Zoom Fit to Page	Displays the entire circuit on the workspace at once.
Zoom to Magnification	Displays the **Zoom** dialog box where you enter the desired magnification.
Zoom Selection	Zooms in on a selected area of the workspace—hold down the mouse button and drag to select an area.
Show Grid	Shows or hides a grid in the background of the circuit window. This helps you place elements in specific locations on a grid.
Show Border	Shows or hides the circuit's border.
Show Page Bounds	Shows or hides page boundaries in the circuit window. This helps you note where circuits will appear on printed output.
Ruler Bars	Shows or hides the ruler bars on the top and left margins that appear outside of the workspace.
Status Bar	Shows or hides the status bar at the bottom of the workspace.
Design Toolbox	Shows or hides the **Design Toolbox**. Refer to the *Design Toolbox* section of Chapter 1, *User Interface*, for more information.
Spreadsheet View	Select/deselect to show/hide the **Spreadsheet View**. Refer to the *Spreadsheet View* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.
Circuit Description Box	Opens a window in which you can enter comments or information about the circuit.
Toolbars	Shows or hides the selected toolbar. Choose to show or hide any or all of the toolbars discussed below. Refer to the *Toolbars* section of Chapter 1, *User Interface*, for more information.
Toolbars»Standard	Select/deselect to show/hide the **Standard** toolbar.
Toolbars»View	Select/deselect to show/hide the **View** toolbar.
Toolbars»Main	Select/deselect to show/hide the **Main** toolbar.

Command	Description
Toolbars»Graphic Annotation	Select/deselect to show/hide the **Graphic Annotation** toolbar.
Toolbars»3D Components	Select/deselect to show/hide the **3D Components** toolbar. You can place virtual 3D components from this toolbar.
Toolbars»Analog Components	Select/deselect to show/hide the **Analog Components** toolbar. You can place virtual analog components from this toolbar.
Toolbars»Basic	Select/deselect to show/hide the **Basic** toolbar. You can place virtual basic components from this toolbar.
Toolbars»Diodes	Select/deselect to show/hide the **Diodes** toolbar. You can place virtual diodes from this toolbar.
Toolbars»Transistor Components	Select/deselect to show/hide the **Transistor Components** toolbar. You can place virtual transistors from this toolbar.
Toolbars»Measurement Components	Select/deselect to show/hide the **Measurement Components** toolbar. You can place virtual measurement components from this toolbar.
Toolbars»Miscellaneous Components	Select/deselect to show/hide the **Miscellaneous Components** toolbar. You can place miscellaneous virtual components from this toolbar.
Toolbars»Components	Select/deselect to show/hide the **Component** toolbar.
Toolbars»Power Source Components	Select/deselect to show/hide the **Power Source Components** toolbar. You can place virtual power source components from this toolbar.
Toolbars»Rated Virtual Components	Select/deselect to show/hide the **Rated Virtual Components** toolbar. You can place rated virtual components from this toolbar.
Toolbars»Signal Source Components	Select/deselect to show/hide the **Signal Source Components** toolbar. You can place virtual signal source components from this toolbar.
Toolbars»Virtual	Select/deselect to show/hide the **Virtual** toolbar, which is used to place virtual components. Refer to the *Placing Virtual Components* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Toolbars»Simulation Switch	Shows or hides the simulation on/off switch.

Command	Description
Toolbars»Simulation	Select/deselect to show/hide the **Simulation** toolbar.
Toolbars»Instruments	Select/deselect to show/hide the **Instruments** toolbar.
Toolbars»Description Edit Bar	Select/deselect to show/hide the **Description Edit Bar**. This toolbar is used to edit the **Circuit Description Box**. Refer to the *Circuit Description Box* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Toolbars»MCU	Select/deselect to show/hide the **MCU** toolbar.
Toolbars»LabVIEW Instruments	Select/deselect to show/hide the **LabVIEW Instruments** toolbar.
Toolbars»NI ELVISmx Instruments	Select/deselect to show/hide the **NI ELVISmx Instruments** toolbar.
Show Comment/Probe	Shows or hides information boxes for selected comments or static probes. Refer to the *Adding a Comment* section of Chapter 2, *Schematic Capture - Basics*, and the *Measurement Probe* section of Chapter 9, *Instruments*, for more information.
Grapher	Shows or hides the **Grapher**, which shows the results of simulation on a graph or chart. Refer to the *Viewing the Analysis Results: Grapher* section of Chapter 10, *Analyses*, for more information.

Place Menu

The following table describes the commands found in the **Place** menu.

Command	Description
Component	Browses the databases (**Master Database**, **Corporate Database** and **User Database**) for components to be placed. Refer to the *Using the Place Component Browser* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Junction	Places a junction where you click. Refer to the *Manually Adding a Junction* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Wire	Places a wire on the workspace that is not connected to any component. Refer to the *Placing Wires Directly Onto Workspace* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Command	Description
Bus	Places a bus with segments created as you click. Refer to the *Buses* section of Chapter 4, *Working with Larger Designs*, for more information.
Connectors	Places various connectors on the workspace.
Connectors»HB/SC Connector	Adds connecting nodes to a circuit for use in a hierarchical block or a subcircuit.
Connectors»Bus HB/SC Connector	Places a bus connector between the main circuit and a hierarchical block or subcircuit. Refer to the *Connecting Buses to HB/SCs* section of Chapter 4, *Working with Larger Designs*, for more information.
Connectors»Off-Page Connector	Places an off-page connector on your workspace. Refer to the *Flat Multi-Sheet Design* section of Chapter 4, *Working with Larger Designs*, for more information.
Connectors»Bus Off-Page Connector	Places a bus connector between the main circuit and a multi-page. Refer to the *Placing a Bus across Multi-Pages* section of Chapter 4, *Working with Larger Designs*, for more information.
New Hierarchical Block	Places a circuit without any components in a hierarchical structure. Refer to the *Adding a Hierarchical Block* section of Chapter 4, *Working with Larger Designs*, for more information.
Hierarchical Block from File	Places a circuit in a hierarchical structure. Refer to the *Placing a HB from an Existing File* section of Chapter 4, *Working with Larger Designs*, for more information.
Replace by Hierarchical Block	Replaces the elements that you have selected with a hierarchical block. Refer to the *Replacing Components with an HB* section of Chapter 4, *Working with Larger Designs*, for more information.
New Subcircuit	Places a new subcircuit without, any components. Refer to the *Adding a Subcircuit* section of Chapter 4, *Working with Larger Designs*, for more information.
Replace by Subcircuit	Replaces the selected components with a subcircuit containing those components. Refer to the *Adding a Subcircuit* section of Chapter 4, *Working with Larger Designs*, for more information.
Multi-Page	Opens a new flat page. Refer to the *Flat Multi-Sheet Design* section of Chapter 4, *Working with Larger Designs*, for more information.

Command	Description
Merge Bus	Merges two buses together. Refer to the *Merging Buses* section of Chapter 4, *Working with Larger Designs*, for more information.
Bus Vector Connect	This is the preferred method for placing numerous connections from a multi-pinned device, such as an IC, to a bus. Refer to the *Bus Vector Connect* section of Chapter 4, *Working with Larger Designs*, for more information.
Comment	"Pins" a comment to the workspace, or directly to a component. When a component with an attached comment is moved, the comment also moves. Refer to the *Adding a Comment* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Text	Places text on the circuit. Refer to the *Adding Miscellaneous Text* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Graphics	Places an assortment of the graphic elements. Refer to the *Graphic Annotation* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Title Block	Places a title block on your workspace. Refer to the *Adding a Title Block* section of Chapter 2, *Schematic Capture - Basics*, for more information.

MCU Menu

For details, refer to Chapter 15, *Multisim MCU*.

Simulate Menu

The following table describes the commands found in the **Simulate** menu.

Command	Description
Run	Runs/resumes the simulation of the active circuit. Running a simulation initializes the circuit, performs the circuit simulation and displays the results on the instrument or grapher.
Pause	Pauses the simulation.
Stop	Stops the simulation.

Command	Description
Instruments»Multimeter	Places a multimeter on the circuit window. The multimeter is used to measure AC or DC voltage or current, resistance, or decibel loss between two nodes in a circuit. Refer to the *Multimeter* section of Chapter 9, *Instruments*, for more information.
Instruments»Function Generator	Places a function generator on the circuit window. The function generator is a voltage source that supplies sine, triangular or square waves. It provides a convenient and realistic way to supply power to a circuit. Refer to the *Function Generator* section of Chapter 9, *Instruments*, for more information.
Instruments»Wattmeter	Places a wattmeter on the circuit window. The wattmeter provides the combined functions of a voltmeter and an ammeter. It is used to measure the magnitude of the active power, that is, the product of the voltage difference and the current flowing through the current terminals in a circuit. Refer to the *Wattmeter* section of Chapter 9, *Instruments*, for more information.
Instruments»Oscilloscope	Places an oscilloscope on the circuit window. The dual-channel oscilloscope displays the magnitude and frequency variations of electronic signals. It can provide a graph of the voltage of one or two signals over time, or allow comparison of one waveform to another. Refer to the *Oscilloscope* section of Chapter 9, *Instruments*, for more information.
Instruments»Four Channel Oscilloscope	Places a four-channel oscilloscope on the circuit window. Refer to the *Four-Channel Oscilloscope* section of Chapter 9, *Instruments*, for more information.
Instruments»Bode Plotter	Places a Bode plotter on the circuit window. A Bode plotter produces a graph of a circuit's frequency response and is useful for analyzing filter circuits. Refer to the *Bode Plotter* section of Chapter 9, *Instruments*, for more information.
Instruments»Frequency Counter	Places a frequency counter on the circuit window. Refer to the *Frequency Counter* section of Chapter 9, *Instruments*, for more information.
Instruments»Word Generator	Places a word generator on the circuit window. A word generator sends digital words or patterns of bits into circuits as a bit stream. Refer to the *Word Generator* section of Chapter 9, *Instruments*, for more information.

Command	Description
Instruments»Logic Analyzer	Places a logic analyzer on the circuit window. A logic analyzer displays the levels of up to 16 digital signals in a circuit. It is used for fast data acquisition of logic states and advanced timing analysis to help design large systems and carry out troubleshooting. Refer to the *Logic Analyzer* section of Chapter 9, *Instruments*, for more information.
Instruments»Logic Converter	Places a logic converter on the circuit window. It can be attached to a circuit to derive the truth table or Boolean expression the circuit embodies, or to produce a circuit from a truth table or Boolean expression. Refer to the *Logic Converter* section of Chapter 9, *Instruments*, for more information.
Instruments»IV Analyzer	Places an IV analyzer on the workspace, which is used to measure current-voltage curves. Refer to the *IV Analyzer* section of Chapter 9, *Instruments*, for more information.
Instruments»Distortion Analyzer	Places a distortion analyzer on the circuit window. A typical distortion analyzer provides distortion measurements for audio signals in the range of 20 Hz to 100 KHz. Refer to the *Distortion Analyzer* section of Chapter 9, *Instruments*, for more information.
Instruments»Spectrum Analyzer	Places a spectrum analyzer on the circuit window. The spectrum analyzer is used to measure frequency versus amplitude. Refer to the *Spectrum Analyzer* section of Chapter 9, *Instruments*, for more information.
Instruments»Network Analyzer	Places a network analyzer on the circuit window. The network analyzer is used to measure the scattering parameters (or S-parameters) of a circuit, commonly used to characterize a circuit intended to operate at higher frequencies. Refer to the *Network Analyzer* section of Chapter 9, *Instruments*, for more information.
Instruments»Agilent Function Generator	Places a simulated Agilent Technologies 33120A Function Generator on the circuit window. This is a high-performance 15 MHz synthesized function generator with built-in arbitrary waveform capability. Refer to the *Agilent Simulated Function Generator* section of Chapter 9, *Instruments*, for more information.
Instruments»Agilent Multimeter	Places a simulated Agilent Technologies 34401A Multimeter on the circuit window. This is a 6 1/2-digit, high-performance digital multimeter. Refer to the *Agilent Simulated Multimeter* section of Chapter 9, *Instruments*, for more information.

Command	Description
Instruments»Agilent Oscilloscope	Places a simulated Agilent Technologies 54622D Oscilloscope on the circuit window. This is a 2-channel +16 logic channels, 100-MHz bandwidth oscilloscope. Refer to the *Agilent Simulated Oscilloscope* section of Chapter 9, *Instruments*, for more information.
Instruments»LabVIEW Instruments	Displays a number of LabVIEW instruments that you can place on the workspace. Refer to the *LabVIEW Instruments* section of Chapter 9, *Instruments*, for more information.
Instruments»Tektronix Oscilloscope	Places a simulated Tektronix TDS 2024 Oscilloscope on the circuit window. Refer to the *Tektronix Simulated Oscilloscope* section of Chapter 9, *Instruments*, for more information.
Instruments»Measurement Probe	Checks voltage and frequency readings on different nodes and pins. Refer to the *Measurement Probe* section of Chapter 9, *Instruments*, for more information.
Instruments»Preset Measurement Probes	Places a preset (static) measurement probe on the workspace. Refer to the *Measurement Probe* section of Chapter 9, *Instruments*, for more information.
Instruments»Current Probe	Places a “clamp-on” current probe on the workspace. Refer to the *Current Probe* section of Chapter 9, *Instruments*, for more information.
Interactive Simulation Settings	Sets default settings for instruments that are based on a transient analysis (such as the oscilloscope, spectrum analyzer and logic analyzer). Refer to the *Interactive Simulation Settings* section of Chapter 9, *Instruments*, for more information.
Digital Simulation Settings	Allows you to choose between optimizing on accuracy or speed of simulation when digital components are included in your circuit. Refer to the *Digital Simulation* section of Chapter 8, *Simulation*, for more information.
Analyses»DC Operating Point	Sets up and runs DC operating point analysis, which determines the DC operating point of a circuit. Refer to the *DC Operating Point Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»AC Analysis	Sets up and runs AC analysis, in which the DC operating point is first calculated to obtain linear, small-signal models for all nonlinear components. Then a complex matrix (containing both real and imaginary components) is created. Refer to the *AC Analysis* section of Chapter 10, *Analyses*, for more information.

Command	Description
Analyses»Transient Analysis	Sets up and runs Transient analysis, also called time-domain transient analysis, which computes the circuit's response as a function of time. Refer to the *Transient Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Fourier Analysis	Sets up and runs Fourier analysis, which evaluates the DC, fundamental and harmonic components of a time-domain signal. Refer to the *Fourier Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Noise Analysis	Sets up and runs Noise analysis, which is used to detect the magnitude of noise power in the output of electronic circuits. Refer to the *Noise Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Noise Figure Analysis	This analysis is part of Multisim's RF Design module. Refer to the *Noise Figure Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Distortion Analysis	Sets up and runs Distortion analysis, which measures harmonic distortion and intermodulation distortion products. Refer to the *Distortion Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»DC Sweep	Sets up and runs DC sweep analysis, which computes the DC operating point of a node in the circuit for various values of one or two DC sources in the circuit. Refer to the *DC Sweep Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Sensitivity	Sets up and runs Sensitivity analysis, which calculates the sensitivity of an output node voltage or current with respect to the parameters of all components (DC sensitivity) or one component (AC sensitivity) in a circuit. Refer to the *DC and AC Sensitivity Analyses* section of Chapter 10, *Analyses*, for more information.
Analyses»Parameter Sweep	Sets up and runs Parameter Sweep analysis, which verifies the operation of a circuit by simulating it across a range of values for a component parameter. Refer to the *Parameter Sweep Analysis* section of Chapter 10, *Analyses*, for more information.

Command	Description
Analyses»Temperature Sweep	Sets up and runs Temperature Sweep analysis, which quickly verifies the operation of a circuit by simulating it at different temperatures. The effect is the same as simulating the circuit several times, once for each different temperature. You control the temperature values. Refer to the *Temperature Sweep Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Pole Zero	Sets up and runs Pole Zero analysis, which finds the poles and zeros in the small-signal AC transfer function of a circuit. Refer to the *Pole Zero Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Transfer Function	Sets up and runs Transfer Function analysis, which calculates the DC small-signal transfer function between an input source and two output nodes (for voltage) or an output variable (for current) in a circuit. It also calculates input and output resistances. Refer to the *Transfer Function Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Worst Case	Sets up and runs Worst Case analysis, a statistical analysis that lets you explore the worst possible effects on circuit performance of variations in component parameters. Refer to the *Worst Case Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Monte Carlo	Sets up and runs Monte Carlo analysis, a statistical analysis that lets you explore how changing component properties affects circuit performance. Refer to the *Monte Carlo Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Trace Width Analysis	Sets up and runs Trace Width analysis, which calculates the minimum trace width needed in a circuit to handle the peak current to be carried by that trace. Refer to the *Trace Width Analysis* section of Chapter 10, *Analyses*, for more information.
Analyses»Batched Analyses	Sets up and runs batched analyses, which let you set up a series of different analyses, or different variations on the same analysis, to be performed on a circuit in sequence. Refer to the *Batched Analyses* section of Chapter 10, *Analyses*, for more information.
Analyses»User Defined Analysis	Sets up and runs a user-defined analysis. This command presents you with a dialog box into which you can type the SPICE commands to be executed to perform the analysis. Refer to the *User Defined Analyses* section of Chapter 10, *Analyses*, for more information.

Command	Description
Analyses»Stop Analysis	Stops the currently running analysis.
Postprocessor	Opens the **Postprocessor** dialog box, which you use to combine the results of several analyses in different ways. To use the **Postprocessor**, you must have performed at least one analysis on your circuit. Refer to Chapter 11, *Postprocessor* for more information.
Simulation Error Log/Audit Trail	Shows or hides the simulation log, which records all the events of a circuit simulation. Refer to the *Simulation Error Help* section of Chapter 8, *Simulation*, for more information about this log.
XSpice Command Line Interface	Opens a window into which you can type Xspice commands to be executed. Refer to the *Simulation from Netlist Without Schematic* section of Chapter 8, *Simulation*, for more information.
Load Simulation Settings	Loads a set of simulation settings that you saved earlier. Refer to the *Loading a Simulation Profile* section of Chapter 8, *Simulation*, for more information.
Save Simulation Settings	Saves a set of simulation settings for later use. Refer to the *Saving a Simulation Profile* section of Chapter 8, *Simulation*, for more information.
Auto Fault Option	Applies faults to randomly selected components in the circuit. You choose the number of faults (either in total, or the number of each type of fault) to be applied.
VHDL Simulation	Runs legacy MultiVHDL simulation software. Refer to the documentation that was included with your copy of MultiVHDL for more information. You must have a copy of MultiVHDL installed for this command to function.
Dynamic Probe Properties	Displays the **Probe Properties** dialog box where you enter the properties for Dynamic Measurement Probes. Refer to the *Dynamic Probe Settings* section of Chapter 9, *Instruments*, for more information.
Reverse Probe Direction	Reverses the polarity of a placed probe. Refer to the *Measurement Probe* section of Chapter 9, *Instruments*, for more information.

Command	Description
Clear Instrument Data	Clears simulation data from instruments like the oscilloscope. Does not clear the **Grapher**.
Use Tolerances	When there is a checkmark beside this command in the **Simulate** menu, the component tolerances set in individual components will be used during simulation. Refer to the *Component Tolerances in Multisim* section of Chapter 8, *Simulation*, for more information about this feature.

Transfer Menu

The following table describes the commands found in the **Transfer** menu.

Command	Description
Transfer to Ultiboard 10	Transfers a design from Multisim to Ultiboard 10. Refer to the *Transferring from Multisim to Ultiboard for PCB Layout* section of Chapter 13, *Transfer/Communication*, for more information.
Transfer to Ultiboard 9 or earlier	Transfers a design from Multisim to Ultiboard 9 or earlier. Refer to the *Transferring from Multisim to Ultiboard for PCB Layout* section of Chapter 13, *Transfer/Communication*, for more information.
Export to PCB Layout	If you are using a PCB layout package other than Ultiboard, you can create files in the necessary formats for transfer to third party layout packages. Refer to the *Transfering to Other PCB Layout Packages* section of Chapter 13, *Transfer/Communication*, for more information.
Forward Annotate to Ultiboard 10	Annotates changes made to a schematic file in Multisim to its existing Ultiboard 10 file. Refer to the *Forward Annotation* section of Chapter 13, *Transfer/Communication*, for more information.
Backannotate from Ultiboard	Backannotates changes made to a circuit in Ultiboard (for example, deleted components) to the Multisim circuit file. Displays a file browser where you choose the backannotation file corresponding to your circuit file. The circuit file must be open before you use this command. Refer to the *Back Annotation* section of Chapter 13, *Transfer/Communication*, for more information.

Command	Description
Forward Annotate to Ultiboard 9 or earlier	Annotates changes made to a schematic file in Multisim to its existing Ultiboard 9 or earlier file. Refer to the *Forward Annotation* section of Chapter 13, *Transfer/Communication*, for more information.
Highlight Selection in Ultiboard	When Ultiboard is running, and you have selected components in Multisim, the corresponding components are highlighted in Ultiboard.
Export Netlist	Exports the netlist of your design. Opens a standard file browser where you can choose the file name and folder for the exported netlist.

Tools Menu

The following table describes the commands found in the **Tools** menu.

Command	Description
Component Wizard	Invokes the **Component Wizard**, which takes you through the steps required to create a component. Refer to the *Creating Components with the Component Wizard* section of Chapter 6, *Component Editing*, for more information.
Database»Database Manager	Displays the Database Manager, where you can add component families, assign button icons to families, edit, copy and delete components, and enter or modify user field titles for those component families. Refer to the *Managing the Database* section of Chapter 5, *Components*, for more information.
Database»Save Component to DB	Saves the selected component, including any changes you made to it, to the database. Refer to the *Saving Placed Components* section of Chapter 5, *Components*, for more information.
Database»Merge Database	Displays the **Merge Database** dialog box where you can merge the contents of another database into your **User** or **Corporate Database**. Refer to the *Merging Databases* section of Chapter 5, *Components*, for more information.
Database»Convert Database	Converts components in an existing **Corporate** or **User Database** into Multisim format. Refer to the *Converting Databases* section of Chapter 5, *Components*, for more information.

Command	Description
Variant Manager	Displays the **Variant Manager** dialog box. Refer to the *Variants* section of Chapter 4, *Working with Larger Designs*, for more information.
Set Active Variant	Displays the **Active Variant** dialog box. Refer to the *Variants* section of Chapter 4, *Working with Larger Designs*, for more information.
Circuit Wizards»555 Timer Wizard	Displays a dialog where you can design astable or monostable timer circuits. Refer to the *555 Timer Wizard* section of Chapter 8, *Simulation*, for more information.
Circuit Wizards»Filter Wizard	Displays a dialog where you can design Low Pass, High Pass, Band Pass and Band Reject filters. Refer to the *Filter Wizard* section of Chapter 8, *Simulation*, for more information.
Circuit Wizards»Op-amp Wizard	Displays a dialog where you can design the following op-amp circuits: Inverting Amplifier; Non-inverting Amplifier; Difference Amplifier; Inverted Summing Amplifier; Non-inverted Summing Amplifier; Scaling Adder. Refer to the *Op-amp Wizard* section of Chapter 8, *Simulation*, for more information.
Circuit Wizards»CE BJT Amplifier Wizard	Displays a dialog where you can design common emitter amplifier circuits by entering the desired specifications into its fields. Refer to the *Common Emitter BJT Amplifier Wizard* section of Chapter 8, *Simulation*, for more information.
Rename/Renumber Components	Renumbers components on the workspace. This is useful if you have numbers that appear all over the workspace, depending on where they were placed, or if you have numbers missing in a sequence. Refer to the *Renaming Component Instances* section of Chapter 4, *Working with Larger Designs*, for more information.
Replace Components	Before using, select the desired component(s) in the circuit window to be replaced. Invokes the **Select a Component** browser from which you can select a new component. Click **OK** to replace the old component(s) with the selected new one. Refer to the *Replacing a Placed Component* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Update Circuit Components	If you open a circuit that was created in an older version of Multisim, you can use this command to update its components to match the current database. Refer to the *Updating Components from Databases* section of Chapter 5, *Components*, for more information.

Command	Description
Update HB/SC Symbols	Updates HB/SC symbols on designs from older versions of Multisim.
Electrical Rules Check	Displays the **Electrical Rules Check** dialog box where you can create reports of electrical connection errors, such as an output pin connected to a power pin. Refer to the *Electrical Rules Checking* section of Chapter 3, *Schematic Capture - Advanced Functions*, for information about running the rule check.
Clear ERC Markers	Clears ERC error markers from the workspace that were placed during earlier ERC runs. Refer to the *Electrical Rules Checking* section of Chapter 3, *Schematic Capture - Advanced Functions*, for information about running the rule check.
Toggle NC Marker	Places a no connection (NC) marker on a selected pin, which prevents wires from being attached to that pin. Refer to the *Marking Pins for No Connection* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Symbol Editor	Creates and edits component symbols. Refer to the *Creating and Editing a Component's Symbol with the Symbol Editor* section for more information.
Title Block Editor	Edits the appearance of, or creates a new, title block. Refer to the *Title Block Editor* section of Chapter 3, *Schematic Capture - Advanced Functions*, for more information.
Description Box Editor	Displays the screen where you edit the contents of the **Circuit Description Box**. Refer to the *Circuit Description Box* section of Chapter 2, *Schematic Capture - Basics*, for more information.
Capture Screen Area	Takes a screen capture of a specific area of your screen. Refer to the *Capturing Screen Area* section of Chapter 2, *Schematic Capture - Basics*, for more information.

Reports Menu

The following table describes the commands found in the **Reports** menu.

Command	Description
Bill of Materials	Use to print a **Bill of Materials** (BOM) for your circuit. A BOM lists the components used in your design and provides a summary of the components needed to manufacture the circuit board. Refer to the *Bill of Materials* section of Chapter 12, *Reports*, for more information.
Component Detail Report	Use to produce a report showing all information stored in the database for a particular component. Refer to the *Component Detail Report* section of Chapter 12, *Reports*, for more information.
Netlist Report	Use to print a report that provides circuit connectivity information for each component. Refer to the *Netlist Report* section of Chapter 12, *Reports*, for more information.
Cross Reference Report	Use to produce a detailed list of all components in the active circuit. Refer to the *Cross Reference Report* section of Chapter 12, *Reports*, for more information.
Schematic Statistics	Use to print a report that lists the quantities of various elements in your circuit, such as real components, virtual components and nets. Refer to the *Schematic Statistics Report* section of Chapter 12, *Reports*, for more information.
Spare Gates Report	Use to produce a report that lists the components in a circuit that have unused gates. Refer to the *Spare Gates Report* section of Chapter 12, *Reports*, for more information.

Options Menu

The following table describes the commands found in the **Options** menu.

Command	Description
Global Preferences	Displays the **Preferences** dialog box. Refer to the *Using the Preferences Dialog Box* section of Chapter 1, *User Interface*, for more information.
Sheet Properties	Displays the **Sheet Properties** dialog box. Refer to the *Using the Sheet Properties Dialog Box* section of Chapter 1, *User Interface*, for more information.
Lock Toolbars	Locks all docked toolbars in their current position. Floating toolbars will be locked in place if you move them to a docked position. This can also be done from the **Toolbars** tab of the **Customize** dialog box. Refer to the *Toolbars tab* and *Locking Toolbars* sections of Chapter 1, *User Interface*, for more information.
Customize User Interface	Tailors your interface to your personal preferences. Refer to the *Customizing the Interface* section of Chapter 1, *User Interface*, for more information.

Window Menu

The following table describes the commands found in the **Window** menu.

Command	Description
New Window	Creates a copy of the existing window.
Close	Closes the active file.
Close All	Closes all open files.
Cascade	Arranges circuit windows so that they overlap.
Tile Horizontal	Resizes all open circuit windows so they all show on the screen in a horizontal orientation. Allows you to quickly scan all open circuit files.

Command	Description
Tile Vertical	Resizes all open circuit windows so they all show on the screen in a vertical orientation. Allows you to quickly scan all open circuit files.
Windows	Displays the **Windows** dialog box where you can show or close the files that you currently have open.
<open files>	Lists the open Multisim circuit files. Select one to make it active.

Help Menu

The following table describes the commands found in the **Help** menu.

Command	Description
Multisim Help	Displays the *Multisim help file*.
Component Reference	Displays Multisim's *Component Reference help file*.
Patents	Displays a file with a list of National Instruments software-related patents.
Release Notes	Displays the NI Circuit Design Suite release notes.
File Information	Displays the **File Information** dialog box.
About Multisim	Displays information on the edition number of Multisim.

B

Archiving Data

National Instruments recommends that you regularly back up the files created within the Multisim and Ultiboard components of NI Circuit Design Suite. Additionally, you should back up internal files that store user-created data, such as database components. This section provides information on where to find these files in order to properly back them up.

NI Circuit Design Suite stores program and user data in two possible locations. Files that need to be accessed on a per-machine basis are stored in different locations, depending on the operating system. Files that are created on a per-user basis are stored in the Windows per-user folder. Multisim and Ultiboard share the same three database files: **Master**, **Corporate**, and **User**.

Backing up the Corporate Database

Windows XP and Windows 2000

The default **Corporate** database location for Windows XP and Windows 2000 is `C:\Documents and Settings\All Users\Application Data\National Instruments\Circuit Design Suite\10.1\database\.`

The file name for the **Corporate** database is `CPCOMP_S.PRJ.`

The `Application Data` folder is a system folder and thus may be hidden.

Complete the following steps to view this folder:

1. Open a standard Windows explorer window.
2. Select **Tools»Folder Options**.
3. Select the **View** tab, **Show hidden files and folders**, **OK**.

Windows Vista

The default **Corporate** database location for Windows Vista is `C:\ProgramData\National Instruments\Circuit Design Suite\10.1\database\.`

The file name for the **Corporate** database is `CPCOMP_S.PRJ`.

The `ProgramData` folder is a system folder and thus may be hidden.

Complete the following steps to view this folder:

1. Open a standard Windows explorer window.
2. Select **Organize»Folder & Search Options**.
3. Select the **View** tab, **Show hidden files and folders**, **OK**.

Note The default location for the **Corporate** database can change if the database has been configured to be shared across a network drive. Check the **Preferences** dialog box to find the location of this database. Refer to the *Preferences - Paths Tab* section of Chapter 1, *User Interface* for more information.

You do not need to back up the **Master** database. However, any user field information that you have added for **Master** dB components is saved in the **Corporate** dB. Consequently, if your corporate database is lost, the **User** field information for **Master** dB components resets to "as shipped".

Also, R, L, and C components found in the **Master** database are in fact a combination of the component's value, component type, tolerance, and footprint manufacturer/type. Whenever a unique combination of these values is selected in the **Select a Component** dialog box, and the **Save unique component on placement** checkbox is enabled, the combination of these values is saved in the **Corporate** database. Consequently, if the **Corporate** dB is lost, these components will also be lost to the **Master** database.

Backing up the User Database and Configuration Files

Windows XP and Windows 2000

For Windows XP and Windows 2000, the **User** database is stored at:

```
C:\Documents and Settings\<User_Name>\Application
Data\National Instruments\Circuit Design
Suite\10.1\database\.
```

The file name for the **User** database is `UsrComp_S_<user name>.usr`.

The user's configuration file is stored at:

```
C:\Documents and Settings\<User_Name>\Application Data\
National Instruments\Circuit Design Suite\10.1\config\.
```

The `Application Data` folder is a system folder and thus may be hidden.

Complete the following steps to view this folder:

1. Open a standard Windows explorer window.
2. Select **Tools»Folder Options**.
3. Select the **View** tab, **Show hidden files and folders**, **OK**.

Windows Vista

For Windows Vista, the **User** database is stored at:

```
C:\Users\<user name>\AppData\Roaming\National
Instruments\10.1\database\.
```

The file name for the **User** database is `UsrComp_S_<user name>.usr`.

The user's configuration file is stored at:

```
C:\Users\<user name>\AppData\Roaming\National
Instruments\10.1\config\.
```

The `AppData` folder is a system folder and thus may be hidden.

Complete the following steps to view this folder:

1. Open a standard Windows explorer window.
2. Select **Organize»Folder & Search Options**.
3. Select the **View** tab, **Show hidden files and folders**, **OK**.

Note You can change the locations of the configuration files and **User** database from the **Preferences** dialog box. Refer to the *Preferences - Paths Tab* section of Chapter 1, *User Interface* for more information.

C

Technical Support and Professional Services

Visit the following sections of the award-winning National Instruments Web site at ni.com for technical support and professional services:

- **Support**—Technical support at ni.com/support includes the following resources:
  - **Self-Help Technical Resources**—For answers and solutions, visit ni.com/support for software drivers and updates, a searchable KnowledgeBase, product manuals, step-by-step troubleshooting wizards, thousands of example programs, tutorials, application notes, instrument drivers, and so on. Registered users also receive access to the NI Discussion Forums at ni.com/forums. NI Applications Engineers make sure every question submitted online receives an answer.
  - **Standard Service Program Membership**—This program entitles members to direct access to NI Applications Engineers via phone and email for one-to-one technical support as well as exclusive access to on demand training modules via the Services Resource Center. NI offers complementary membership for a full year after purchase, after which you may renew to continue your benefits.

    For information about other technical support options in your area, visit ni.com/services, or contact your local office at ni.com/contact.
- **Training and Certification**—Visit ni.com/training for self-paced training, eLearning virtual classrooms, interactive CDs, and Certification program information. You also can register for instructor-led, hands-on courses at locations around the world.
- **System Integration**—If you have time constraints, limited in-house technical resources, or other project challenges, National Instruments Alliance Partner members can help. To learn more, call your local NI office or visit ni.com/alliance.

If you searched ni.com and could not find the answers you need, contact your local office or NI corporate headquarters. Phone numbers for our worldwide offices are listed at the front of this manual. You also can visit the Worldwide Offices section of ni.com/niglobal to access the branch office Web sites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

Index

Numerics

555 Timer Wizard, 8-16

A

ABM modeling, 6-47
AC analysis, 10-35
AC Motor model maker, 6-57
AC sensitivity analysis, 10-62
Add a Footprint dialog, 6-52
Add Buslines dialog box, 4-21
Add Variant Name dialog box, 4-38
adding
    components, 6-3
    device/model parameters, 10-24
adding traces to a graph, 10-20
Advanced Pin Mapping dialog, 6-53
Agilent Simulated Function Generator, 9-63
Agilent simulated instruments, 9-63
Agilent Simulated Multimeter, 9-65
Agilent Simulated Oscilloscope, 9-66
algorithm
    diagonal Gmin stepping, 8-11
    execution order, 8-13
    source stepping, 8-12
ammeter
    about, 9-73
    connecting, 9-74
    mode, 9-74
    multimeter measurement options, 9-8
    resistance, 9-74
analog component creation, 6-3
analog ground, 4-5
analyses
    about, 10-1
    AC, 10-35
    AC sensitivity, 10-62
    analysis options tab, 10-28
    Analysis Parameters tab, 10-22
    audit trail, 8-24
    batched, 10-113
    custom options, 10-117
    cutting/copying/pasting pages, graphs, and charts, 10-16
    DC operating point, 10-30
    DC sensitivity, 10-62
    DC sweep, 10-57
    determining component use, 3-6
    distortion, 10-50
    Fourier, 10-40
    incomplete, 10-29
    matching network, 14-23
    Monte Carlo, 10-93
    nested sweep, 10-113
    noise, 10-44
    output tab, 10-23
    parameter sweep, 10-67
    performing (general instructions), 10-22
    pole zero, 10-87
    printing graphs and charts, 10-19
    results, 8-24
    RF, 14-20
    RF Characterizer, 14-20
    summary tab, 10-29
    temperature sweep, 10-74
    trace width, 10-107
    transfer function, 10-76
    transient, 10-37
    user-defined, 10-115
    viewing charts, 10-15
    viewing graphs, 10-7
    working with pages, 10-6
    worst case, 10-82
Analysis Expression dialog box, 10-26

analysis options, 10-28
analysis options tab, about, 10-28
analysis output, manipulating, 11-1
Analysis Parameters tab, about, 10-22
assigning faults to components, 3-8
audit trail, 8-24
Auto Fault option, 3-9
auto-backup, 1-21
automatic wiring, 2-20
Automation API
    introduction, 16-1
axes, Bode plotter settings, 9-21

B

back annotation, 13-5
backing up projects, 4-53
batched analyses, 10-113
Bill of Materials, 12-1
BJT model maker, 6-58
Bode plotter
    about, 9-19
    axes settings, 9-21
    magnitude, 9-21
    phase, 9-21
    readouts, 9-22
    settings, 9-21
BOM report
    using, 12-3
Boolean expressions, entering, 9-34
Boost Converter model maker, 6-70
breaking, 15-31
breakpoints, 15-29
browsing database, 2-2
Buck Boost Converter model maker, 6-71
Buck Converter model maker, 6-71
build errors and warnings, 15-21
building MCU projects, 15-24
bus
    placing, 4-15
Bus Entry Connection dialog box, 4-23, 4-25
Bus Merge dialog box, 4-22
Bus Properties dialog box, 4-21
Bus Vector Connect dialog, 4-26
bus width, 1-27
Buses
    placing, 4-18
Button Appearance dialog, 1-35

C

capturing screen area, 2-41
CE BJT Wizard, 8-20
changing component value/model, 3-2
channel settings, 9-17
chart
    properties, 10-15
charts
    cut/copy/paste, 10-16
    printing, 10-19
    using in postprocessor, 11-8
    viewing, 10-15
checking electrical connections, 3-34
circuit
    adding instruments to, 9-2
    consistency, checking in simulation, 8-6
    equation, 8-9
    printing files, 2-58
    simulation mechanisim, 8-7
circuit window
    multiple, 2-1
    placing components, 2-3
clock
    logic analyzer, 9-29
Clock Setup dialog, 9-29
code model
    about, 6-94
    creating, 6-95
    implementation file, 6-106
    interface file, 6-99

color
component, 2-19
customizing, 1-25
selecting, 1-25
wire, 2-26
command line, 8-6
component
properties, General tab, 6-17
symbol, editing, 6-17
value/model, changing, 3-2
Component Detail Report
dialog box, 12-5
Component Detail report
using, 12-5
Component Editing
about, 6-1
Component List dialog, 6-42
component tolerances, 8-3
component value
capacitors, 3-3
inductors, 3-3
resistors, 3-3
component wizard, 6-3
component's properties dialog, 2-18
Variant tab, 4-42
components
about, 5-1
adding, 6-3
assigning faults, 3-8
Auto Fault option, 3-9
changing value/model, 3-2
classification in database, 5-2
color, 2-19
copying, 5-7
copying placed, 2-18
copying symbols, 6-19
creating model, 6-39
detail report, 12-5
determining use in analyses, 3-6
displaying information, 3-1
editing, 6-16
editing model, 6-39
editing symbol, 6-17
finding, 2-29
flipping, 2-28
general properties, 6-17
information stored, 5-4
moving, 2-16
package information, 6-49
pins, 6-49
placed, properties, 3-1
placing, 2-2
placing on circuit window, 2-3
removing, 5-7
rotating, 2-28
user fields, 6-56
virtual, 2-2
wiring, 2-19
configuration files
back-up, B-2
connecting
ammeter, 9-74
voltmeter, 9-73
controlled source syntax, 6-47
conventions used in the manual, *v*
Convergence Assistant, 8-26
Convert Database dialog, 5-12
converting V6 database, 5-12
converting V7 database, 5-12
copying
component symbol, 6-19
components, 5-7
placed components, 2-18
corporate database
back-up, B-1
creating
components, 6-3
projects, 4-50
Cross Reference report, 12-11
using, 12-11

Cuk Converter model maker, 6-71
current probe, 9-74
Current Probe Properties dialog, 9-74
custom analysis options, 10-117
Custom Analysis Options dialog, 10-117
Customize dialog
    Commands tab, 1-33
    Keyboard tab, 1-34
    Menu tab, 1-34
    Options tab, 1-35
    pop-up menus, 1-35
    Toolbars tab, 1-33
customizing interface, 1-19
cut/copy/paste pages, graphs, and charts, 10-16

D

database
    adding a component family, 5-9
    browsing for components, 5-3
    component classification, 5-2
    deleting a component family, 5-9
    finding components, 5-3
    levels, 5-2
    loading a component toolbar button, 5-9
    management, 5-6
    searching for components, 5-3
    selecting components, 2-2
    structure of, 5-1
    V6 conversion, 5-12
    V7 conversion, 5-12
Database Information dialog box, 5-12
database levels, 5-2
Database Manager dialog, 5-6
Date and Time dialog, 2-45
DC convergence assistance, 8-11
DC operating point analysis, 10-30
DC sensitivity analysis, 10-62
DC sweep analysis, 10-57
debug window settings, 15-27
default analysis, postprocessor, 11-7
Delete Multi-Page dialog, 4-2
deleting
    component family, 5-9
    components, 5-7
    empty component families, 5-9
Description Label dialog, 2-50
Description window, 2-42
description, adding to circuit, 2-42
Design Toolbox
    Hierarchy tab, 4-48
diagnostic tools (NI resources), C-1
diagonal Gmin stepping, 8-11
difference amplifier, 8-21
digital component creation, 6-7
digital ground, 4-5
diode model maker, 6-72
displaying information about placed components, 3-1
distortion analysis, 10-50
Distortion Analyzer
    Settings dialog, 9-34
distortion analyzer, 9-34
    SINAD, 9-35
documentation
    conventions used in the manual, *v*
    NI resources, C-1
drivers (NI resources), C-1
dropping a two-pinned component onto a wire, 2-15
dual-channel oscilloscope, 9-15
Duplicate Component Name dialog, 5-12
dynamic probe properties, 9-37

E

Edit footprint dialog, 3-5
Edit Forms dialog box, 2-54
Edit Labels dialog, 2-51
Edit menu, A-4
Edit Model dialog, 3-4

editing
component symbol, 6-17
component toolbar button, 5-9
components, 6-16
Electrical Rules Checking, 3-34
Electronic Parameters, 6-56
ERC markers, clearing, 3-38
ERC Options tab, 3-37
ERC Rules tab, 3-39
error log/audit trail, 8-24
examples (NI resources), C-1
Excel, exporting simulation results to, 13-5
exporting
simulation results to Excel, 13-5
simulation results to MathCAD, 13-5
exporting a netlist, 13-8
exporting components from the dB manager, 5-8
exporting simulation results, 13-5
exporting to Excel, 13-6
exporting to MathCAD, 13-5

F

faults, assigning to components, 3-8
File menu, A-1
File Properties dialog, 4-52
files
locking and unlocking, 4-52
opening with project, 4-51
removing from project, 4-50
filter nodes, 10-24
Filter Wizard, 8-18
Find Component dialog, 2-29
Finding components, 2-29
finding components, 2-29
finding components in database, 5-3
Flat multi-sheet design, 4-1
flipping components, 2-28
font settings, 1-27
Footprint tab, 6-49
forms
about, 2-54
forward annotation, 13-4
Four Channel Oscilloscope, 9-43
Fourier analysis, 10-40
frequency, 9-26
Frequency Counter, 9-50
function generator
about, 9-11
signal options, 9-13

G

general component properties, 6-17
General tab, 6-17
global nets
0, 4-5
analog ground, 4-5
digital ground, 4-5
GND, 4-5
VCC, 4-5
VDD, 4-5
VEE, 4-5
VSS, 4-5
Graph properties
axis tabs, 10-12
general tab, 10-7
traces tab, 10-13
Grapher, 10-2
Grapher window, 10-2
Graphic Annotation, 1-7, 2-38
Graphic Annotation Toolbar, 1-7
graphs
cut/copy/paste, 10-16
printing, 10-19
using in post processor, 11-8
viewing, 10-7
grounding oscilloscope, 9-17

H

help, technical support, C-1
hidden pins, 2-19
Hierarchical Block
    creating new, 4-7
Hierarchical Block Properties dialog, 4-7
Hierarchy tab
    Variants folder, 4-48
horizontal Bode plotter settings, 9-21

I

Ideal Transformer (mult. winding) model maker, 6-75
implementation file (code model), 6-106
importing
    files, other formats, 13-8
    SPICE, Cadence® PSpice® netlist, 13-10
In Use list, 2-14
incomplete analyses, 10-29
information stored for components, 5-4
in-place edit mode, 6-28
Insert Object dialog, 2-46
instrument drivers (NI resources), C-1
instruments
    adding to a circuit, 9-2
    ammeter, 9-73
    Bode plotter, 9-19
    distortion analyzer, 9-34
    function generator, 9-11
    logic analyzer, 9-26
    logic converter, 9-31
    measurement options, 9-8
    multimeter, 9-7
    multiple, 9-4
    oscilloscope, 9-15
    spectrum analyzer, 9-36
    using, 9-3
    voltmeter, 9-72
    wattmeter, 9-13
    word generator, 9-23
integration order, maximum, 8-10
interactive components, 8-3
interactive simulation, 8-2
Interactive Simulation Settings, 9-5
interdigital model, 14-40
interface
    customizing, 1-19
    elements, 1-1
    toolbars, 1-1
interface file (code model), 6-99
internal settings of multimeter, 9-11
inverting amplifier, 8-21
inverting summing amplifier, 8-21
IV Analyzer, 9-53
    Simulate Parameters dialog, 9-57

J

junction, adding, 2-27

K

KnowledgeBase, C-1

L

labeling, 2-31
labels
    assigning to nets, 2-32
LabVIEW instruments, 9-78
    creating, 9-83
    microphone, 9-80
    signal analyzer, 9-81
    signal generator, 9-82
    speaker, 9-81
    streaming signal generator, 9-82
    toolbar, 9-83
levels
    of database, 5-2
Linear Transformer (mult. winding) model maker, 6-76

Linear Transformer (neutral terminal) model maker, 6-76
loading external hex file project, 15-19
loading model, 6-45
locking files, 4-52
locking net names in place, 2-32
locking toolbars, 1-11
logic analyzer
about, 9-26
adjusting clock, 9-29
reset, 9-28
stop, 9-28
triggering, 9-30
logic converter, 9-31
constructing a truth table, 9-33
deriving truth table from a circuit, 9-32
entering Boolean expressions, 9-34
lossy line model, 14-39

M

machine code file simulation, 15-21
magnitude, 9-21
managing families, 5-9
managing files in MCU workspace, 15-7
manual wiring, 2-22
matching network analyses, 14-23
MathCAD, exporting simulation results to, 13-5
maximum integration order, 8-10
MCU Code Manager
file management, 15-10
overview, 15-8
MCU design overview, 15-4
MCU project build settings, 15-11
MCU Wizard, 15-5
MCU workspace, 15-20
measurement options
ammeter, 9-8
ohmmeter, 9-8
Measurement Probe, 9-37
dynamic connection, 9-37, 9-40
placed connection, 9-41
settings, 9-37
static connection, 9-37, 9-41
memory view, 15-32
menu
Edit, A-4
File, A-1
Options, A-23
Place, A-9
pop-up, 1-12
Reports, A-22
Simulate, A-11
Symbol Editor, 6-28
Tools, A-19
Transfer, A-18
View, A-6
Window, A-23
microstrip line model, 14-35
microstrip open end model, 14-36
mode
ammeter, 9-74
voltmeter, 9-73
model
changing component, 3-2
loading, 6-45
model creation
using code modeling, 6-94
Model Data report, 12-10
using, 12-10
model maker
AC Motor, 6-57
BJT, 6-58
Boost Converter, 6-70
Buck Boost Converter, 6-71
Buck Converter, 6-71
Cuk Converter, 6-71
diode, 6-72
Ideal Transformer (mult. winding), 6-75

Linear Transformer (mult. winding), 6-76
Linear Transformer (neutral terminal), 6-76
Non-linear Transformer (mult. winding), 6-77
silicon controlled rectifier, 6-87
Two Winding Linear Transformer, 6-76
module
RF, 14-1
Monte Carlo analysis, 10-93
moving component, 2-16
multimeter, 9-7
internal settings of, 9-11
signal mode, 9-10
using ammeter measurement option, 9-8
using ohmmeter measurement option, 9-8
Multimeter Settings dialog, 9-11
multiple circuit windows, 2-1
multiple instruments, 9-4
multiple traces, postprocessor, 11-8

N

National Instruments support and services, C-1
nested sweep analysis, 10-113
Net dialog, 2-32
netlist errors, 8-22
Netlist report, 12-6
using, 12-6
netlist warnings, 8-22
nets, assigning labels, 2-32
network analyzer, 14-17
New Family Name dialog, 5-9
New Project dialog, 4-50
NI support and services, C-1
noise analysis, 10-44
Noise Figure Analysis, 14-28
non-inverting amplifier, 8-21
non-inverting summing amplifier, 8-21
Non-linear Transformer (mult. winding) model maker, 6-77

O

Off-page connector
placing, 4-1
ohmmeter, 9-8
Op-amp Wizard, 8-21
opening
debug view, 15-26
ewb5 files, 13-9
file within project, 4-51
project, 4-51
ulticap files, 13-9
Options menu, A-23
options, analysis, 10-117
oscilloscope
about, 9-15
channel settings, 9-17
grounding, 9-17
time base, 9-16
trigger, 9-18
output tab, about, 10-23

P

package information, 6-49
Page Name dialog, 4-1
page properties, 10-6
pages
cut/copy/paste, 10-16
using in analyses, 10-6
using in postprocessor, 11-8
Paragraph dialog, 2-44
parameter sweep analysis, 10-67
Paste Special dialog, 2-18
PCB layout, transfer to, 13-3
phase, 9-21
pin information, 6-49
pin parameters, 6-48

pins, adding to symbols, 6-23
Place menu, A-9
placed components, properties, 3-1
Placed Probe, 9-37
placing
    multi-page, 4-1
placing a bus, 4-18
placing capacitors, 2-5
placing components, 2-2
placing inductors, 2-5
placing multi-section components, 2-8
placing potentiometers, 2-5
placing resistors, 2-5
placing variable capacitors, 2-5
placing variable inductors, 2-5
pole zero analysis, 10-87
pop-up
    from selected wire, 1-17
    from title block, 1-18
pop-up menu, 1-12
    from selected component, 1-15
    no component selected, 1-12
postprocessor
    about, 11-1, 11-2, 11-9
    basic steps, 11-2
    creating multiple traces, 11-8
    functions, 11-9
    pages, graphs and charts, 11-8
    using the default analysis, 11-7
    variables, 11-9
Power Pro
    using code modeling, 6-94
pre-defined fields in database, 5-5
preferences
    General tab, 1-23
    Preferences dialog, 1-20
    Sheet Properties dialog, 1-24
    Simulation tab, 1-23
Preferences Dialog
    Parts tab, 1-22
    Paths tab, 1-20
    Save tab, 1-21
Print Circuit Setup dialog, 2-58
Print Instruments dialog, 9-5
printing
    circuit files, 2-58
    graphs and charts, 10-19
    instruments, 9-5
Probe Properties dialog, 9-37
programming examples (NI resources), C-1
project
    backing up, 4-53
    creating, 4-50
    opening, 4-51
    removing files from, 4-50
    saving, 4-50
Project bar, 4-50
project management, about, 4-50
Projects
    working with, 4-51
Properties dialog, 2-46
properties of placed components, 3-1

Q

questions
    creating, 2-54
    linking to a circuit, 2-54
    setting submit options, 2-56

R

readouts, 9-22
Reference Designator Prefix Setup dialog, 4-13
Refine Search Component dialog, 5-3
removing components, 5-7
Rename Busline dialog box, 4-21
Rename Variant dialog box, 4-38

replacing components with a subcircuit, 4-10
reports
    Bill of Materials, 12-1
    component detail, 12-5
Reports menu, A-22
resistance
    ammeter, 9-74
    voltmeter, 9-72
RF analyses, 14-20
RF characterizer analysis, 14-20
RF components
    about, 14-2
    interdigital model, 14-40
    lossy line model, 14-39
    microstrip line model, 14-35
    microstrip open end model, 14-36
    RF model makers, 14-33
    RF spiral inductor model, 14-36
    strip line model, 14-37
    stripline bend model, 14-38
    waveguide model, 14-34
RF instruments, 14-9
    network analyzer, 14-17
    spectrum analyzer, 14-9
RF module, 14-1
    about, 14-1
    components *See also* RF components
    instruments *See also* RF instruments
RF simulation, 8-14
RF tutorial, 14-41
RLC Components tab, 5-11
rotating components, 2-28
Rule Check tab, 3-39

S

saving
    project, 4-50
saving placed components, 5-8
scaling adder, 8-21
schematic capture, 2-1
Schematic Statistics report, 12-7
    using, 12-8
Search Component dialog, 5-3
Search Component Result dialog, 5-3
Select a Component browser, 2-18
    other buttons, 2-11
    using, 2-3
Select a Footprint dialog, 6-49
Select a Form dialog, 2-56
Select a Model dialog, 6-41
Select Model Data dialog, 6-46
Select Model Maker dialog, 6-42
Select variables to save dialog, 10-25
Set Parameters dialog, 6-41, 6-49
settings, oscilloscope channel, 9-17
Sheet Properties
    Circuit tab, 1-25
Sheet Properties dialog
    Font tab, 1-27
    PCB tab, 1-28
    Visibility tab, 1-29
    Wiring tab, 1-27
signal mode, 9-10
signal options, 9-13
silicon controlled rectifier model maker, 6-87
Simulate menu, A-11
simulation
    checking circuit consistency, 8-6
    circuit, 8-7
    circuit equation, 8-9
    diagonal Gmin stepping, 8-11
    interactive, 8-2
    maximum integration order, 8-10
    RF, 8-14
    source stepping, 8-12
    stages of, 8-7
    starting and stopping, 8-4
simulation error help, 8-26
simulation errors, 8-22, 8-24

simulation errors and warnings
troubleshooting, 9-6
simulation markers, 15-29
simulation speed, 8-5
Simulation tab, 8-22, 8-24
Simulation toolbar, 1-4
SINAD, 9-35
software (NI resources), C-1
source code file, 15-24
source stepping, 8-12
Spare Gates report, 12-9
using, 12-9
spectrum analyzer, 9-36, 14-9
SPICE reference
AM modulation source, 7-54
analog devices, 7-18
arbitrary source, 7-60
BJT, 7-37
BST modulation source, 7-55
capacitor, 7-20
circuit parameters, 7-7
comments, line continuation, 7-9
compatibility modes, 7-70
coupled mutual inductor, 7-23
current controlled current source, 7-65
current controlled voltage source, 7-63
current-controlled switch, 7-36
diode, 7-24
documentation conventions, 7-1
exponential source, 7-52
FM modulation source, 7-54
FSK modulation source, 7-55
general purpose syntax, 7-2
independent current source, 7-58
independent voltage source, 7-47
inductor, 7-22
JFET, 7-29
lossless transmission line, 7-26
lossy transmission line, 7-27
mathematical expressions, 7-10
mathematical functions, operators, constants, 7-11
MESFET, 7-31
modulation source, 7-53
MOSFET, 7-42
noise source, 7-57
number format, 7-8
piecewise linear file source, 7-53
piecewise linear source, 7-52
polynomial specifications, 7-62
primitive device declarations, 7-2
pulse source, 7-50
resistor, 7-19
single frequency FM source, 7-51
sinusoidal source, 7-51
SPICE subcircuits, 7-4
SWP modulation source, 7-56
uniform R.C. Line, 7-28
unusual device syntax, 7-69
user-defined functions, 7-17
voltage controlled current source, 7-64
voltage-controlled switch, 7-34
voltage-controlled switch, smooth transition, 7-35
voltge controlled voltage source, 7-61
XARB source, 7-57
XSPICE code model, 7-68
XSPICE syntax reference, 7-68
SPICE simulation
circuit, 8-7
circuit equation, 8-9
diagonal Gmin stepping, 8-11
maximum integration order, 8-10
source stepping, 8-12
spiral inductor, RF, 14-36
Spreadsheet view, 3-10
buttons, 3-15
Components tab, 3-12
Nets tab, 3-10
PCB Layers tab, 3-14
Simulation tab, 3-14, 8-22, 8-24

static probe, 9-41
static probe properties, 9-38
stepping, 15-31
strip line model, 14-37
stripline bend model, 14-38
structure of database, 5-1
subcircuit
    replacing components with, 4-10
Subcircuit Name dialog, 4-10
summary tab, about, 10-29
support, technical, C-1
symbol
    adding pins, 6-23
    copying, 6-19
    editing, 6-17
Symbol Editor
    Default Pin Options dialog, 6-25
    Draw Tools toolbar, 6-35
    Drawing toolbar, 6-38
    Edit menu, 6-29
    Enter Text dialog, 6-27
    File menu, 6-28
    Graphics menu, 6-31
    Help menu, 6-33
    in-place edit mode, 6-28
    Layout menu, 6-32
    menus, 6-28
    palette, 6-33
    Pin Array Options dialog, 6-26
    Pins menu, 6-31
    Place Pin toolbar, 6-37
    pop-up menus, 6-33
    screen, 6-19
    Spreadsheet view, 6-21
    Standard toolbar, 6-33
    symbol pins, 6-23
    symbol shape, 6-23
    Tool menu, 6-32
    toolbars, 6-33
    using, 6-19
    View menu, 6-30
    working with, 6-23
    Zoom toolbar, 6-35
system toolbar, 1-2

T

Tabs dialog, 2-45
technical support, C-1
Tektronix simulated instruments, 9-69
Tektronix Simulated Oscilloscope, 9-69
temperature sweep analysis, 10-74
time base, 9-16
Title Block
    adding, 2-33
    contents, editing, 2-33
Title Block Editor, 3-16
    Draw Tools toolbar, 3-30
    Drawing toolbar, 3-32
    Edit menu, 3-23
    Enter Text dialog, 3-19
    Enter Title Block Attribute dialog, 3-19
    field codes, 3-21
    Fields menu, 3-25
    File menu, 3-23
    Graphics menu, 3-27
    Help menu, 3-28
    menus, 3-23
    pop-up menus, 3-28
    Spreadsheet View, 3-22
    Standard toolbar, 3-29
    Toolbars, 3-28
    Tools menu, 3-28
    View menu, 3-24
    Zoom toolbar, 3-30
toolbar sizing, 1-12
toolbars, 1-1
    system, 1-2
    zoom, 1-2
Tools menu, A-19
trace width analysis, 10-107

training and certification (NI resources), C-1
transfer function analysis, 10-76
Transfer menu, A-18
transfer to other PCB Layout, 13-3
transfer to Ultiboard PCB Layout, 13-2
transfer/communication functions, introduction to, 13-1
transferring data, 13-1
transferring, from Multisim to Ultiboard, 13-2
transferring, multi-section components to PCB layout, 13-3
transient analysis, 10-37
Trigger Settings dialog, 9-30
trigger, oscilloscope, 9-18
triggering
  logic analyzer, 9-30
  word generator, 9-26
troubleshooting (NI resources), C-1
troubleshooting simulation, 9-6
truth table
  constructing, 9-33
  deriving from circuit, 9-32
Two Winding Linear Transformer model maker, 6-76
two-pinned components
  dropping directly onto a wire, 2-15

U

Ultiboard, transfer to, 13-2
Ulticap Import dialog, 13-8
unlocking files, 4-52
Updating components, 5-14
user database
  back-up, B-2
user field titles, modifying, 5-10
user fields, 5-5
  working with, 6-56
User Fields tab, 3-7
user interface
  customizing, 1-19
  elements, 1-1
  toolbars, 1-1
user-defined analysis, 10-115
using instruments, 9-3
using model makers, 6-57

V

value, changing component, 3-2
variables, postprocessor, 11-9
Variant Manager dialog, 4-38
Variant tab, 4-42
variants, 4-37
  Variant Manager dialog, 4-38
Variants Filter dialog box, 12-12
Variants folder, 4-48
version control
  about, 4-50
  using, 4-53
Version Control dialog, 4-53
vertical Bode plotter settings, 9-21
VHDL component creation, 6-11
View menu, A-6
virtual components, 2-2
virtual wiring, 2-27
voltmeter
  about, 9-72
  connecting, 9-73
  mode, 9-73
  resistance, 9-72

W

wattmeter, 9-13
waveguide model, 14-34
Web resources, C-1
Window menu, A-23

wire
color, 2-26
re-shaping, 2-25
width, 1-27
wiring
“mid-air”, 2-24
automatic, 2-20
combining automatic and manual, 2-23
components, 2-19
directly onto workspace, 2-24
manual, 2-22
properties, 2-24
setting options, 1-27
virtual, 2-27
width, 2-24
Wizard
555 Timer, 8-16
Commom Emitter BJT, 8-20
Component, 6-3
Filter, 8-18
Op-amp, 8-21
word generator, 9-23
controls, 9-25
entering words, 9-24
setting clock frequency, 9-26
triggering, 9-26
using word patterns in, 9-25
worst case analysis, 10-82

X

Xspice
command line, 8-6
XSpice Command Line dialog, 8-6

Z

zoom toolbar, 1-2